



**Almerindo de Oliveira
Paiva**

**Projecto e Construção de um Medidor de Oxigénio em
Gases**



**Almerindo de Oliveira
Paiva**

**Projecto e Construção de um Medidor de Oxigénio em
Gases**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações (Mestrado Integrado), realizada sob a orientação científica do Dr. António Ferreira Pereira de Melo, Professor Catedrático do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

O júri

Presidente

Professor Doutor. Alexandre Manuel Moutela Nunes da Mota

Professor Associado do Departamento de Electrónica Telecomunicações e Informática da Universidade de Aveiro

Vogais

Professor Doutor Pedro António Amado Assunção

Professor Coordenador da Escola Superior de Tecnologia e Gestão do Instituto Politécnico de Leiria

Professor Doutor António Ferreira Pereira de Melo

Professor Catedrático do Departamento de Electrónica Telecomunicações e Informática da Universidade de Aveiro (Orientador)

Agradecimentos

Quero aproveitar esta circunstância para manifestar os meus agradecimentos a todos os Professores que ao transmitiram os seus conhecimentos, ao longo destes anos, me permitiram concluir mais uma etapa da minha formação académica.

Ao Professor Catedrático António Ferreira Pereira de Melo, um agradecimento especial pela sua orientação e conhecimentos transmitidos ao longo da minha formação académica.

Quero agradecer ao Departamento de Ambiente e Ordenamento, particularmente ao Professor Luís António da Cruz Tarelho, pela sua disponibilidade e por me autorizar a frequentar as suas instalações, sem as quais não teria sido possível a realização deste projecto.

À Selmatron por ter financiado parte dos componentes necessários à realização deste trabalho.

Quero agradecer ao meu pai e à minha mãe, Mário e Hermínia, pelos esforços e dificuldades ultrapassadas para eu chegar mais longe.

Agradeço também à minha namorada, Carla, pelo apoio e carinho prestado nos últimos anos da minha formação académica.

Por fim, mas não menos importante, agradeço aos meus familiares, amigos e colegas pela entreatajuda e amizade, permitindo-me superar algumas etapas mais complicadas.

Palavras-chave

Combustão industrial, Poluição atmosférica, Rendimento de uma caldeira, Medidor de percentagem de Oxigénio e Sonda lambda.

Resumo

As normas governamentais de protecção ambiental são cada vez mais rigorosas, o que implica que sejam tomadas medidas de controlo na emissão de gases. As grandes indústrias de combustão utilizam analisadores de combustão muito dispendiosos que envolvem tecnologias de ponta. Porém nas pequenas indústrias que envolvem processos de combustão, como sejam, caldeiras, fornalhas, etc. estes analisadores não são economicamente viáveis. Sucede que mesmo os analisadores existentes no mercado destinados ao controlo de processos de combustão de dimensão reduzida são muito caros, razão pela qual muitos construtores de caldeiras optam por não empregar qualquer tipo de controlo da combustão e, consequentemente, da emissão de gases poluentes.

O presente trabalho visa contribuir para a solução deste problema de natureza económica e ambiental. O seu desenvolvimento ocorreu numa parceria entre o Departamento de Electrónica, Telecomunicações e Informática e a firma Selmatron sediada em Aveiro.

O projecto desenvolvido visava a construção de um medidor de oxigénio simples, de fácil utilização e manutenção e, simultaneamente, de baixo custo em relação às unidades existentes no mercado.

Numa primeira fase, dedicámos a nossa atenção ao estudo do modo como o controlo eficiente do excesso de ar (quantidade de ar acima da mínima necessária para se dar uma combustão completa) que entra no processo de combustão, permite otimizar o rendimento térmico e reduzir a emissão de componentes poluentes. Foram abordadas algumas tecnologias utilizadas para controlar a eficiência térmica e foi feito um estudo sobre a sonda lambda.

Numa segunda fase, foram equacionadas diferentes versões conducentes a um possível produto final.

Keywords

Industrial Combustion, Atmospheric pollution, Boiler efficiency, Oxygen percentage measuring, Lambda Sensor

Abstract

Environmental protection regulations issued by governments are becoming increasingly stringent, what implies measures to control the emission of polluting gases and particles. The major combustion industries use very expensive combustion analyzers and they invest heavily in cutting-edge technology. However in small industries, involving boilers, furnaces, etc. these gas analyzers are not economically viable. It happens that even the existing measurement equipment intended for small combustion process are so expensive that most of the boiler manufacturers choose do not to use any controller at all to regulate the emission gases.

The present work aims to contribute for the solution of this environmental and economical problem. Its development occurred under partnership between the Electronics, Telecommunications and Informatics Department of Aveiro University and the Aveiro's company Selmatron.

The present project intended the construction of a reliable simple low priced oxygen meter, in comparison to the existing market products.

To start, we have been examining how the effective control of the excess air over the amount of the air necessary for a completed combustion to take place, can optimize the thermal efficiency and reduce emission of pollutants. We approached some technologies used to control the excess air and we studied how the lambda probe works.

In a second phase, we approached different solutions versions leading to a final product.

Índice de Conteúdos

| | | |
|--------|---|----|
| 1. | Introdução | 1 |
| 1.1. | Enquadramento..... | 1 |
| 1.2. | Organização do documento | 1 |
| 2. | Introdução Teórica e Estado da Arte..... | 3 |
| 2.1. | Introdução | 3 |
| 2.2. | Excesso de ar – conceito chave para controlar os níveis de emissão de poluentes e o rendimento de uma caldeira | 3 |
| 2.2.1. | Influência do excesso de ar no rendimento da combustão | 4 |
| 2.2.2. | Influência do excesso de ar nos níveis de poluição ambiental | 4 |
| 2.2.3. | Valor óptimo de excesso de ar | 4 |
| 2.2.4. | Métodos de controlo do excesso de ar numa caldeira. | 6 |
| 2.3. | Sonda Lambda | 8 |
| 2.4. | Princípio de funcionamento da Sonda Lambda | 14 |
| 3. | Visão global da Estrutura do equipamento a desenvolver - especificação e decomposição em blocos funcionais | 17 |
| 3.1. | Introdução | 17 |
| 3.2. | Especificações do sistema medidor da percentagem de oxigénio num gás | 17 |
| 3.3. | Blocos funcionais do sistema | 17 |
| 3.3.1. | Descrição das tarefas a exercer por cada bloco funcional..... | 18 |
| 3.4. | Determinação do método a utilizar pelo algoritmo a implementar no microcontrolador para calcular a percentagem de oxigénio..... | 19 |
| 3.4.1. | Preparação do ensaio | 19 |
| 3.4.2. | Esquema de montagem | 20 |
| 3.4.3. | Procedimento de ensaio | 21 |
| 3.4.4. | Resultados | 22 |
| 3.4.5. | Análise dos resultados..... | 22 |
| 3.4.6. | Conclusão | 23 |
| 4. | Hardware..... | 25 |
| 4.1. | Introdução | 25 |
| 4.2. | Determinação do número de bits necessário para o conversor analógico-digital . | 25 |
| 4.3. | Determinação do número de bits necessário para o conversor digital-analógico . | 26 |
| 4.4. | Bloco de Transmissão Analógica do valor de %O ₂ | 26 |
| 4.5. | Bloco de aquisição e processamento de dados | 27 |

| | | |
|--------|---|----|
| 4.6. | Bloco de acondicionamento de sinal | 29 |
| 4.6.1. | Primeira solução – Amplificação + Offset. | 29 |
| 4.6.2. | Segunda solução – Amplificação variável + offset variável..... | 31 |
| 4.7. | Alimentação dos diferentes blocos funcionais. | 35 |
| 4.8. | Interligação entre os diferentes blocos..... | 36 |
| 4.8.1. | Interligação entre o bloco de acondicionamento e o bloco de aquisição e processamento de dados | 36 |
| 4.8.2. | Interligação entre bloco de aquisição e processamento de dados e o bloco de transmissão analógica da %O ₂ | 36 |
| 4.8.3. | Interligação microcontrolador – PC | 39 |
| 4.9. | Placa PCB | 41 |
| 5. | Software | 43 |
| 5.1. | Introdução | 43 |
| 5.2. | MPLAB®ICD 2 In-Circuit Debugger/Programmer | 43 |
| 5.3. | Fluxograma da aplicação desenvolvida..... | 45 |
| 5.4. | Frequência de amostragem | 46 |
| 6. | Resultados | 47 |
| 6.1. | Introdução | 47 |
| 6.2. | Alterações no esquema de montagem para a realização dos testes..... | 47 |
| 6.2.1. | Equipamento utilizado | 48 |
| 6.3. | Apresentação de resultados..... | 49 |
| 7. | Solução Final – Introdução da auto-calibração por software na aplicação desenvolvida 53 | |
| 7.1. | Introdução | 53 |
| 7.2. | Calibração do Medidor de %O ₂ por software | 53 |
| 7.2.1. | Alterações ao Hardware..... | 54 |
| 7.2.2. | Software | 54 |
| 7.3. | Resultados obtidos | 57 |
| 8. | Conclusões e Trabalho futuro | 61 |
| | Bibliografia: | 63 |

Índice de Figuras

| | |
|---|----|
| Figura 2-1– Relação coeficiente de excesso de ar – níveis de poluição..... | 5 |
| Figura 2-2 – Produtos da combustão à saída da caldeira | 7 |
| Figura 2-3 – Instrumento de medida universal de percentagem de oxigénio LT1 | 10 |
| Figura 2-4 – Instrumento de medida universal de percentagem de oxigénio LT2 | 12 |
| Figura 2-5 – Instrumento de medida universal de percentagem de oxigénio LT10 | 13 |
| Figura 2-6 – Analisador de medição dos gases de combustão EGR 5230 | 13 |
| Figura 2-7 – Princípio de funcionamento da sonda lambda | 15 |
| Figura 3-1 – Conversão dum sinal logaritmico num sinal linear | 18 |
| Figura 3-2 – Blocos funcionais do Medidor a desenvolver | 18 |
| Figura 3-3 – Fotomontagem do esquema de montagem do equipamento utilizado nos ensaios com a sonda lambda | 20 |
| Figura 4-1 – AD420..... | 27 |
| Figura 4-2 – Esquema eléctrico base do AD420..... | 27 |
| Figura 4-3 – Esquema eléctrico base do PIC24HJ12GP201 | 28 |
| Figura 4-4 – Esquema eléctrico da Primeira solução para o circuito de acondicionamento de sinal | 29 |
| Figura 4-5 – Esquema eléctrico da Segunda solução para o circuito de acondicionamento de sinal | 33 |
| Figura 4-6 – Esquema de alimentação dos diferentes blocos do Medidor..... | 35 |
| Figura 4-7 – Esquema eléctrico de ligação do circuito de acondicionamento ao PIC24HJ12GP201 | 36 |
| Figura 4-8 – Diagrama de tempo dos sinais SPI imposta pela AD420..... | 38 |
| Figura 4-9 – Esquema eléctrico de ligação PIC24HJ12GP201 – AD420..... | 39 |
| Figura 4-10 – Esquema eléctrico de ligação PIC – Conector RJ45..... | 39 |
| Figura 4-11 – Esquema eléctrico de ligação cabo RJ45 – MAX232 – Ficha fêmea DB9 | 40 |
| Figura 4-12 – Top view da placa com a MAX232 incorporada na ficha fêmea DB9 | 40 |
| Figura 4-13 – Bottom view da placa com a MAX232 incorporada na ficha fêmea DB9 | 40 |
| Figura 4-14 – Placa PCB do Medidor | 41 |
| Figura 5-1 – Esquema de ligação PC – MPLAB ICD2 – Placa com microcontrolador | 43 |
| Figura 5-2 – Fluxograma do código desenvolvido para o Medidor..... | 45 |
| Figura 6-1 – Esquema de montagem com as novas alterações | 48 |
| Figura 7-1 – Placa PCB com o botão de calibração | 54 |
| Figura 7-2 – Fluxograma do código do Medidor com calibração por software | 55 |
| Figura 7-3 – Consulta dos dados de memória do microcontrolador nos endereços onde residem os parâmetros M e B | 58 |

Índice de Gráficos

| | |
|---|----|
| Gráfico 3.1 – Resposta da sonda em função da percentagem de oxigénio | 22 |
| Gráfico 3.2 – Discrepância entre os valores obtidos pelo analisador e os valores calculados a partir da equação 3.1 | 23 |
| Gráfico 4.1 – Avaliação da reprodutividade da sonda lambda | 30 |
| Gráfico 4.2 – Resposta da sonda à saída do primeiro andar de ganho..... | 34 |
| Gráfico 6.1 – Resposta da sonda à saída do primeiro andar de ganho em função da percentagem de oxigénio..... | 49 |
| Gráfico 6.2 – Valores obtidos pelo Medidor e pelo analisador..... | 50 |
| Gráfico 6.3 – Discrepância entre os valores indicados pelo Medidor face aos valores indicados pelo analisador | 50 |
| Gráfico 7.1 – Resposta da sonda em função da percentagem de oxigénio | 58 |
| Gráfico 7.2 – Valores obtidos pelo Medidor e pelo analisador..... | 59 |
| Gráfico 7.3 – Discrepância entre os valores indicados pelo Medidor face aos valores indicados pelo analisador | 59 |

Índice de Tabelas

| | |
|--|----|
| Tabela 4.1 – Alimentações dos integrados constituintes do sistema | 35 |
| Tabela 4.2 – Tabela com os diferentes modos de validação de dados: polaridade e fase | 37 |

Acrónimos

%O₂ – Percentagem de oxigénio
%EGR – Exhaust Gas Recirculation (percentagem de gás efluente de recirculação)
ADC – Analog-Digital Converter (Conversor analógico-digital)
CO – Monóxido de carbono
CPHA – Fase SPI
CPOL – polaridade SPI
DAC – Digital-Analog Converter (Conversor digital-analógico)
DAO – Departamento de Ambiente e Ordenamento da Universidade de Aveiro
dsPIC – Digital Signal Controllers
F/F – Fêmea-Fêmea
H₂ – Hidrogénio (molécula)
H₂SO₄ – Ácido sulfúrico
I/O – entrada/saída
ICD – In-Circuit Debugger
ICE – In-Circuit Emulators
ICSP – In Circuit Serial Programming
Ksps – Kilosample per second
LSB - Least Significant Bit (bit menos significativo)
M/M – Macho-Macho
Msps – Megasample per second
N₂ – Azoto ou Nitrogénio
NO_x – Óxidos de azoto
O₂ – Oxigénio (molécula)
O²⁻ – Iões de oxigénio
PC – Computador
PIC – Programmable Interrupt Controller
PTC – Positive Temperature Coefficient (Coeficiente de temperatura positivo)
R – coeficiente de correlação
SCK – Serial Clock (SPI)
SDI – Serial Data In (SPI)
SDO – Serial Data OUT (SPI)
SO₂ – Dióxido de enxofre
SO₃ – Trióxido de enxofre
SO_x – Óxidos de enxofre
SPI – Serial Peripheral Interface (Interface periférica série)
SRAM – Static Random Access Memory (Memória Estática de Acesso Aleatório)
SS – Slave Select
UART – Universal Synchronous Receiver/Transmitter
V_{eff} – tensão eficaz
V_{rms} – Tensão eficaz
Y₂O₃ – Óxido de ytrio
ZrO₂ – Dióxido de zircónio

Capítulo 1

1. Introdução

1.1. Enquadramento

Esta dissertação descreve o estudo desenvolvido no Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro, em parceria com a firma Selmatron, e tem por objectivo desenvolver um medidor da percentagem de oxigénio existente numa mistura gasosa. Este medidor deve ser económico, simples, eficiente, com custos de manutenção bastante reduzidos e de fácil calibração. O principal objectivo a atingir com este equipamento é medir a percentagem de oxigénio dos gases de combustão de caldeiras e fornalhas, tendo em vista o controlo do rendimento energético e da emissão de poluentes pelas mesmas.

1.2. Organização do documento

Esta dissertação está estruturada em 8 capítulos, de acordo com a seguinte disposição:

O capítulo 2 aborda a importância do excesso de ar¹ no controlo do rendimento da combustão e do volume de emissão de poluentes. Apresenta-se a faixa de excesso de ar ideal, que permite otimizar o rendimento e cumprir as normas sobre a emissão de gases poluentes, assim como é feito o levantamento relativo aos métodos e às tecnologias actualmente utilizadas no controlo da combustão industrial. Por último, analisa o estado da arte sobre a tecnologia baseada na sonda lambda, e o princípio de funcionamento desta.

O capítulo 3 fornece uma descrição geral do Medidor a desenvolver. As especificações do projecto são aqui listadas. É exposta a solução idealizada para “atacar” o problema, baseada em três grandes blocos funcionais. Por fim, são apresentados os primeiros resultados obtidos com a sonda lambda que foram essenciais para desenvolver o *hardware* e o *software* do medidor da percentagem de oxigénio.

¹ Excesso de Ar – é a quantidade de ar que entra no sistema de combustão para além da quantidade teoricamente necessária para que a combustão seja completa (ar estequiométrico). A quantidade de ar além da quantidade estequiométrica assegura um teor de oxigénio suficiente até ao final da chama, o que permite superar as deficiências de mistura combustível/ar do queimador.

O capítulo 4 descreve o *hardware* desenvolvido para implementar os blocos funcionais enunciados no capítulo 3.

O capítulo 5 apresenta o código, em linguagem C, desenvolvido para ser implementado no microcontrolador. É feita uma breve descrição sobre a ferramenta de programação utilizada, nomeadamente o MPLAB ICD2 da Microchip.

O capítulo 6 discute os resultados conseguidos com a solução desenvolvida. As conclusões obtidas são aqui apresentadas.

O capítulo 7 apresenta uma solução mais robusta, que permite otimizar o tempo de calibração do medidor desenvolvido com a introdução da calibração via software. São ainda apresentados os resultados obtidos com este método.

Por fim, no capítulo 8, são apresentadas as conclusões finais do projecto, assim como alguns aspectos a desenvolver num trabalho futuro.

Capítulo 2

2. Introdução Teórica e Estado da Arte

2.1. Introdução

A crescente preocupação com o meio ambiente obriga os governos a tomarem medidas de protecção ambiental cada vez mais rigorosas e a imporem normas relativas ao teor dos gases poluentes produzidos nas combustões industriais (CO^2 , NOx^3 , SOx^4). Por outro lado, é do interesse dos proprietários das caldeiras industriais, fornalhas, etc., que o rendimento (razão entre a quantidade de energia térmica gerada e a quantidade de combustível introduzido) das mesmas seja o mais elevado possível, de modo a reduzir os custos de exploração.

No que se segue, explica-se o modo como o excesso de ar influencia tanto o rendimento térmico, como os níveis de emissão de poluentes. Discute-se, nomeadamente, a faixa de excesso de ar ideal que cumpre as normas ambientais relativamente à emissão de poluentes e que optimiza o rendimento de uma caldeira. É ainda feito um levantamento sobre os métodos e tecnologias actualmente utilizadas no controlo de um processo de combustão. Por fim, é apresentado o estudo realizado sobre a sonda lambda.

2.2. Excesso de ar – conceito chave para controlar os níveis de emissão de poluentes e o rendimento de uma caldeira

Antigamente, o controlo do rendimento térmico da caldeira e o controlo da poluição ambiental eram realizados separadamente. Contudo, alguns estudos revelam que existe uma relação entre o rendimento térmico e o nível de emissão de poluentes das caldeiras, envolvendo o excesso de ar [1]. Segundo este estudo, o controlo eficiente do excesso de ar permite optimizar a eficiência térmica, assegurando ao mesmo tempo uma diminuição da emissão de poluentes e o cumprimento das normas ambientais.

² CO – monóxido de carbono.

³ NOx – óxidos de Azoto.

⁴ SOx – óxidos de Enxofre.

2.2.1. Influência do excesso de ar no rendimento da combustão

Do ponto de vista do rendimento térmico da combustão, o excesso de ar é o factor mais determinante, pois controla o volume, a temperatura e a entalpia dos produtos da combustão⁵. Um excesso de ar elevado é indesejável, porque diminui a temperatura da chama e aumenta as perdas de calor devido à entalpia dos gases efluentes, o que reduz a eficiência térmica. Um excesso de ar baixo conduz a uma combustão incompleta, à formação de CO, fuligem e fumaça, além de possibilitar a acumulação de combustível não queimado, causando risco de explosão. [1]

2.2.2. Influência do excesso de ar nos níveis de poluição ambiental

Do ponto de vista ecológico, há um valor de excesso de ar para o qual o nível de NOx é máximo. Diminuindo o excesso de ar a partir desse valor, diminui-se na câmara de combustão a quantidade de O₂ que não reagiu com o combustível, e a quantidade de azoto (N₂), responsáveis pela formação de NOx. Consequentemente, o teor de NOx diminui. Aumentando o excesso de ar a partir desse valor, o teor de NOx também diminui devido à diminuição da temperatura da chama⁶.

A diminuição das emissões de NOx é acompanhada de um aumento de emissões de fuligem, cenosferas⁷ e coque⁸. Ao diminuir a emissão de NOx pela redução do excesso de ar, há um aumento da emissão de fuligem.

O teor de CO diminui rapidamente com o excesso de ar. Contudo, aumentando o excesso de ar, aumenta a quantidade de O₂ disponível para a oxidação do SO₂ em SO₃, aumentando a formação de ácido sulfúrico (H₂SO₄) [1]

2.2.3. Valor óptimo de excesso de ar

A escolha do valor óptimo de excesso de ar, que maximiza o rendimento e minimiza a poluição, exige um estudo especial.

A figura 2.1 [1] descreve o modo como as emissões de CO e NOx variam com o aumento do coeficiente de excesso de ar⁹, e permite-nos ajudar a determinar a faixa de coeficiente de excesso de ar óptimo que devemos utilizar.

⁵ A entalpia dos produtos de combustão – é o calor libertado durante as reacções de combustão.

⁶ A formação de NOx pode resultar de duas fontes: devido a reacções com o nitrogénio do combustível (NOx combustível) ou devido à reacção do azoto com oxigénio (NOx térmico). A formação de NOx térmico só ocorre a temperaturas muito elevadas (superiores a 1500°C).

⁷ Cenoferas – partículas de carbono muito finas visíveis nos gases pelo fumo mais ou menos denso, que se formam na sua maioria na zona primária de combustão.

⁸ Coque – é um tipo de combustível derivado do carvão betuminoso. O coque obtém-se do aquecimento da hulha (ou carvão betuminoso), sem combustão, num recipiente fechado.

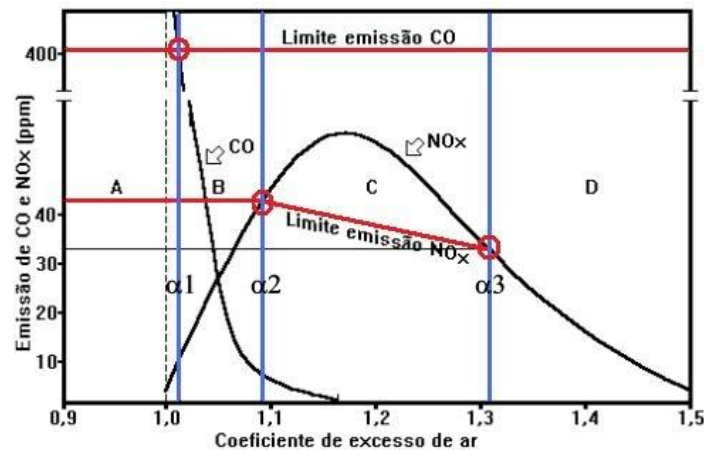


FIGURA 2-1- RELAÇÃO COEFICIENTE DE EXCESSO DE AR – NÍVEIS DE POLUIÇÃO

Como se pode observar, para um coeficiente de excesso de ar inferior a α_1 , o teor de CO excede os limites permitidos por lei. Para os valores de coeficiente de excesso de ar compreendidos entre α_2 e α_3 , o teor de NOx também excede os limites estipulados na lei. Para operar nesta última faixa, teria que se recorrer a equipamentos de tratamento dos gases efluentes. Contudo, a eficiência da combustão é inferior à da faixa compreendida entre α_1 e α_2 , pelo que esta faixa é a ideal, uma vez que os teores de CO e NOx estão dentro dos limites admitidos por lei. Além dos factores ecológicos, esta faixa de coeficiente de excesso de ar apresenta um rendimento elevado.

Todavia, para operar nesta faixa, é necessário que a combustão seja realizada em multi-estágios, combustão sub-estequiométrica ou recirculação de gases.

Para valores de coeficiente de excesso de ar superior a α_3 , os teores de CO e NOx cumprem os limites impostos, contudo, esta faixa é a que apresenta menor rendimento, uma vez que este decresce devido ao arrefecimento da chama.

O valor de coeficiente de excesso de ar (entre α_1 e α_2) que maximiza o rendimento térmico, depende da composição química do combustível, das dimensões e da estrutura da caldeira. Assim, o coeficiente de excesso de ar deve ser ajustado experimentalmente para cada caso, de modo a obter-se o rendimento máximo.

De acordo com o exposto, o controlo da combustão deve ser realizado através do controlo da quantidade de ar de combustão, uma vez que a energia gerada pela caldeira deixa de estar dependente da quantidade de combustível introduzido, o que aumenta o rendimento da caldeira.

Alguns sistemas de combustão apresentam um comportamento diferente e por essa razão algumas das faixas citadas podem não existir.

⁹ Coeficiente de excesso de ar – também conhecido por coeficiente lambda, é um modo de expressar a relação ar/combustível e é a razão entre a quantidade total de ar utilizada na combustão e a quantidade de ar estequiométrica.

2.2.4. Métodos de controlo do excesso de ar numa caldeira.

Tradicionalmente, a quantidade de ar que alimenta a combustão numa caldeira é regulada em função da quantidade de combustível ou de vapor de água por accionamento de uma válvula de ar – controlo em malha aberta. Porém, este tipo de controlo estabelece uma relação ar/combustível fixa, muitas vezes insatisfatória e não toma em consideração que, nos processos industriais, a composição, a temperatura e a viscosidade dos combustíveis, assim como, a temperatura do ar, estão sujeitos a variações e requerem reajustes com frequência. [1]

As caldeiras mais recentes já incorporam um sistema de controlo em malha fechada. Esse controlo pode ser realizado antes, durante ou depois da combustão. No primeiro caso (antes da queima), uma parcela de combustível é desviada para um calorímetro, que determina a composição química do combustível, e possibilita estabelecer a relação ar/combustível mais apropriada para a combustão. Contudo, apesar da sua elevada eficiência, este método torna-se muito dispendioso e só se justifica ser empregue em caldeiras industriais muito grandes, que investem pesadamente na tecnologia de ponta. [2]

No segundo caso (durante a queima), é utilizado um sensor de ionização que fornece uma corrente eléctrica, correlacionando a temperatura da chama ou a relação ar/combustível com a eficiência da combustão. No entanto, a faixa de operação desta tecnologia está limitada pelo calor gerado durante a combustão. Esta tecnologia encontra-se em fase de estudo e desenvolvimento. [2]

No terceiro caso (após a queima), a relação ar/combustível é determinada a partir da medição da concentração de O_2 , CO e CO_2 nos produtos da combustão, à saída da caldeira. A figura 2.2 [3] apresenta a variação da concentração de alguns produtos da combustão com o coeficiente de excesso de ar.

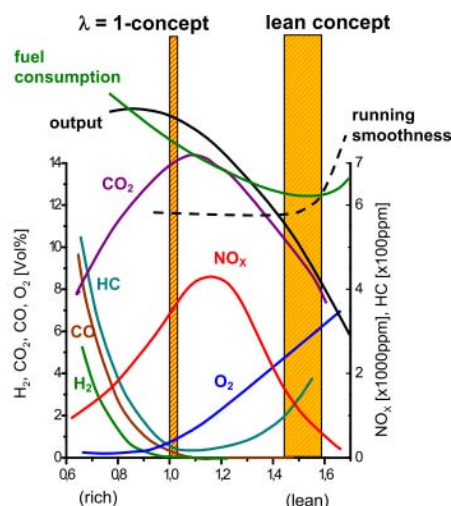


FIGURA 2-2 – PRODUTOS DA COMBUSTÃO À SAÍDA DA CALDEIRA¹⁰

O método baseado na medição da concentração de CO₂, requer o ajuste do set-point para cada combustível, e por ser mais caro que o analisador de O₂, geralmente não é recomendado. [1]

Os equipamentos analisadores de oxigénio são os mais utilizados no controlo da combustão, devido ao seu baixo custo, tempo de resposta, baixa manutenção e porque permitem medir directamente¹¹ o excesso de ar na chaminé. Tal como o analisador de CO₂, o analisador de O₂ também requer o ajuste do set-point para cada combustível e para cada taxa de combustão. Além disso, este método só se torna viável na ausência de infiltrações de ar provenientes do exterior, pois estas irão afectar a concentração de oxigénio, causando erros de leitura elevados. [1]

A tecnologia baseada na medição de CO apresenta-se como a mais vantajosa, pelo facto da formação do CO só ocorrer, quando a quantidade de ar for insuficiente para completar a combustão, e assim o set-point não depende da composição química do combustível. Como o teor de CO nos gases resultantes da combustão é muito reduzido, as infiltrações de ar não interferem nos valores medidos, dado que a diluição do ar atmosférico com os gases de combustão é insignificante. Contudo, estes analisadores devem ser desligados durante mudanças de carga e operações a baixa carga, uma vez que pequenas variações de carga podem causar grandes variações do

¹⁰ Teoricamente, a percentagem de oxigénio presente nos gases de combustão deveria ser nula para um coeficiente de excesso de ar entre 0 e 1. Contudo, na prática verifica-se que a percentagem de oxigénio nessa gama é não nula e que a combustão numa fornalha nunca é completa, devido à cinética química (variação temporal da concentração de reagentes) e à heterogeneidade da composição do combustível que oferece resistência à transferência de calor e de massa.

¹¹ Na ausência de infiltrações de ar provenientes do exterior, a quantidade de oxigénio presente nos gases efluentes corresponde à quantidade de oxigénio em excesso não utilizada na combustão. Deste modo, a quantidade de oxigénio medida nos gases de combustão permite determinar directamente o excesso de ar.

valor medido. Além disso, queimadores sujos ou uma mistura deficiente fazem aumentar o excesso de ar, logo reduzem a eficiência da combustão. [1]

Neste trabalho optou-se pelo desenvolvimento da tecnologia que tem por base, a medição da concentração de oxigénio presente nos gases de combustão para determinação do excesso de ar, com a finalidade de reduzir os níveis de poluição emitidos para a atmosfera e, também, com a finalidade de tornar a combustão mais rentável. A razão pela qual se optou por esta tecnologia, deve-se ao seu baixo custo (o custo dum sensor de oxigénio da Bosch – sonda lambda – ronda os 250€ [4]), ao seu curto tempo de resposta, ao tempo de vida relativamente longo (superior a 10.000h \approx 1 ano e 2 meses [5]) e aos reduzidos custos de manutenção da sonda lambda, quando comparado com os monitores de combustão que custam entre 2000€ a 20000€ (Junho 2007) [2], e que exigem manutenção anual, com trocas de células, que podem apresentar custos entre 400€ a 1200€ (Junho 2007) [2], dependendo do tipo de células saturadas, necessitando de posterior recalibração. A estas despesas de manutenção, acresce ainda os prejuízos causados pelo tempo em que a caldeira está fora de operação, quando o monitor de combustão tiver que ser enviado para a assistência técnica autorizada dos fabricantes destes monitores.

Além disso, o controlo da eficiência da relação ar/combustível, com base na tecnologia da sonda lambda, é geralmente utilizado nos automóveis, razão pela qual é uma tecnologia que está bem desenvolvida. Para a sua utilização nas caldeiras industriais é apenas necessário, fazer uma adaptação da faixa de funcionamento deste sensor. Não considerámos a tecnologia que correlaciona a temperatura da chama com o excesso de ar, por esta ainda estar em fase de estudo. Também não optámos pelos analisadores de CO₂, devido ao seu elevado custo (quando comparado com o custo da sonda lambda). A razão pela qual também não se optou pelos analisadores de CO deve-se ao facto de não poderem estar em operação permanente (têm de ser desligados durante as mudanças de carga) e ao facto de exigirem limpeza frequente, a fim de não serem introduzidos erros de leitura.

2.3. Sonda Lambda

A sonda lambda foi inventada em 1976 pela empresa alemã Robert Bosch GmbH. No mesmo ano, essa tecnologia foi aplicada pela primeira vez nos automóveis suecos da Volvo. Os automóveis Volvo da série 240 e 260, que possuíam a sonda lambda, foram exportados para a Califórnia, que entretanto introduziu normas para o controlo de emissão de CO₂. [6]

Até há bem pouco tempo, a sonda era exclusivamente aplicada na indústria automóvel para reduzir a emissão de poluentes e melhorar o desempenho dos motores dos automóveis. Nos últimos anos, em consequência da crescente preocupação com o meio ambiente e face às medidas adoptadas a nível mundial para combater a poluição ambiental, começaram a surgir, na Alemanha, EUA e Suécia, os primeiros estudos e as primeiras aplicações destes sensores voltados para a indústria [2].

O primeiro Medidor de combustão, especialmente desenvolvido para as aplicações indústrias, com base nesta sonda, foi desenvolvido pela empresa alemã Lamtec (1998) [2]. Desde 1999, vários investigadores nesta área, procuraram usar a sonda lambda em aplicações de combustão industrial, nomeadamente, para medir o teor de O_2 na faixa de 12 a 15% em caldeiras industriais (Unéus et al.), para monitorizar a combustão em leito fluídizado¹² (Wiesendorf et al.), para determinar experimentalmente o excesso de ar utilizando o número de Wobbe¹³ o que, por sua vez, é determinado através desta sonda [2]. Em 2007, Lutero et al. testaram e comprovaram experimentalmente a viabilidade da sonda lambda como um medidor da % O_2 , em aplicações de combustão industrial [2].

Verifica-se que, actualmente, já existem no mercado algumas empresas que fabricam medidores de % O_2 , baseados nas sondas lambda. Assim, importa analisar os produtos já existentes no mercado no que se refere aos custos e funcionalidades.

Uma das empresas existentes no mercado dos medidores de teor de oxigénio, baseado na sonda lambda, é a já atrás citada empresa alemã Lamtec. Esta tem à disposição dos potenciais interessados três produtos, nomeadamente, Lambda-Transmitter LT1 + O_2 -probe LS 1, Lambda-Transmitter LT2 + O_2 -probe LS 2 e Lambda-Transmitter LT10.

A sonda lambda LS1 consiste numa célula electroquímica, projectada de um modo particular, para produzir uma corrente eléctrica que é proporcional à concentração de O_2 . O modo como foi projectada confere-lhe atributos especiais, como por exemplo, a fácil calibração e a insensibilidade às variações da temperatura, dentro da gama de funcionamento. A sonda pode ser calibrada usando apenas o ar atmosférico, uma vez que a curva característica desta é uma recta que passa pela origem. Apesar de se manter insensível à variação da temperatura, esta limita a sua gama de funcionamento. Quanto maior for a temperatura, menor é a gama de funcionamento

¹² Leito fluidizado – é uma tecnologia de combustão que permite a redução das emissões de gases, por isso requer uma menor necessidade de sistemas de controlo de poluição do ar.

¹³ Número de Wobbe - representa o calor fornecido pela queima de gases combustíveis através de um orifício submetido a pressões constantes, a montante e a jusante desse orifício.

[7]. O preço deste sensor ronda os 1972€ (Preço obtido via email da Lamtec em Abril de 2009).

O Lambda-Transmitter LT1, apresentado na figura 2.3, é um instrumento de medida universal de %O₂ baseado num microprocessador e deve ser usado em conjunto com a sonda LS1. As características mais relevantes deste instrumento são [8]:

- Apresentar um tempo de estabilização inferior a 15 segundos;
- Requerer baixa manutenção;
- Apresentar boa precisão, melhor que 0,2% de O₂ na gama de 0 a 21% de O₂;
- Apresentar resultados insensíveis à temperatura;
- Não requerer gases padrão de calibração (calibra automaticamente com o ar atmosférico);
- Poder ser utilizado em aplicações cuja temperatura do gás de combustão chegue aos 1700°C;
- Medir directamente o gás de exaustão, não necessitando de purificação do gás.

O LT1 com uma saída analógica de 4-20mA ou 0-10V para indicar o valor da %O₂ da mistura gasosa, sem display gráfico e sem as funcionalidades opcionais, tem um custo mínimo de 3109€. A incorporação dum display gráfico tem um custo de 760€. (Preço obtido via email da Lamtec em Abril de 2009)

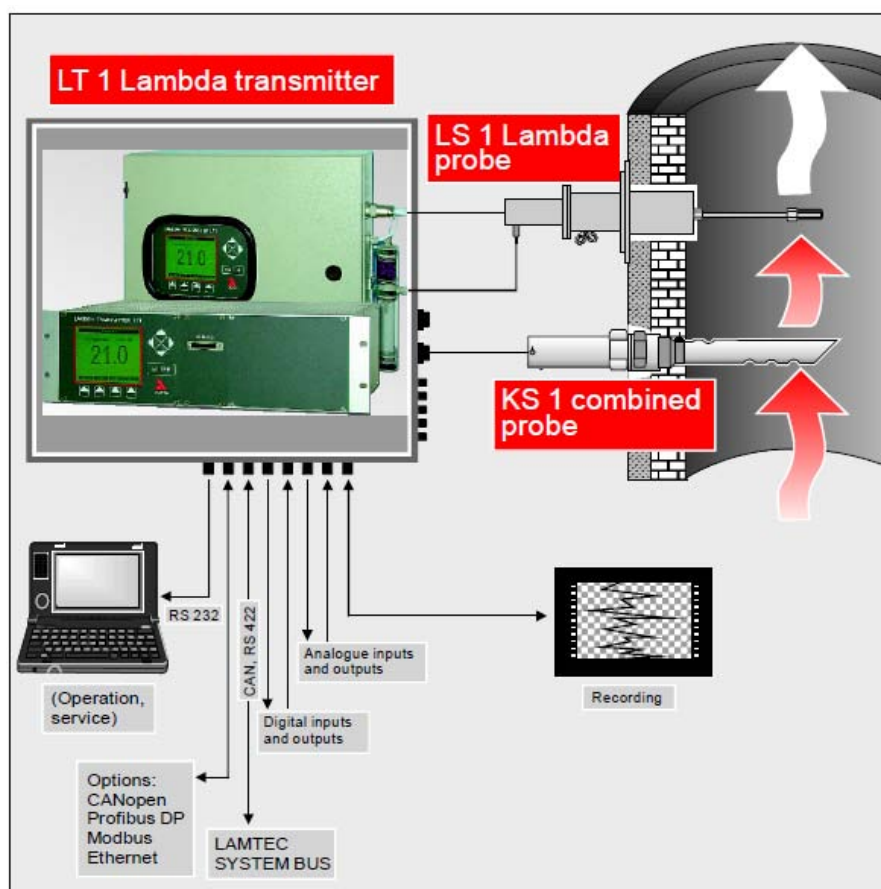


FIGURA 2-3 – INSTRUMENTO DE MEDIDA UNIVERSAL DE PERCENTAGEM DE OXIGÉNIO LT1

A sonda lambda LS2 consiste numa célula electroquímica de dióxido de Zircónio, cujo princípio de funcionamento é igual ao da sonda lambda da Bosch que foi utilizado no desenvolvimento deste projecto e que será explicado mais à frente. A resposta da sonda está, preferencialmente, adequada para as baixas concentrações de oxigénio, pois as medidas tornam-se mais precisas, à medida que o teor de oxigénio diminui, como acontece nas sondas da Bosch. Não se devem utilizar gases de concentração de oxigénio elevada para calibrar a sonda, como por exemplo, o ar atmosférico, porque a resposta da sonda está sujeita a erros maiores nestes níveis de concentração [9]. O preço deste sensor importa os 530€ (Preço obtido via email da Lamtec em Abril de 2009).

O Lambda-Transmitter LT2, apresentado na figura 2.4, é um instrumento de medida universal de teor de O_2 baseado num microprocessador que, em conjunto com a sonda LS2, mede a concentração de O_2 dos gases de combustão. Possui ainda as seguintes funcionalidades (opcionais) [8]:

- Calcula a eficiência de combustão;
- Determina a temperatura dos gases;
- Calcula a concentração de CO_2 quando lhe for adicionado a sonda KS1 da Lamtec;
- Detecta a acumulação de resíduos por queimar (CO/H_2).

As principais características são [8]:

- Apresentar um tempo de estabilização inferior a 15 segundos;
- Medir directamente o gás de exaustão, não necessitando de purificação do gás;
- Não requerer gases de calibração;
- Consumir entre 15 a 25 watt;
- Requerer baixa manutenção.

Uma das desvantagens deste conjunto é que a temperatura máxima está limitada a 300°C, e a precisão é relativamente fraca, ou seja, $\pm 10\%$ do valor lido.

O preço mínimo do LT2 com uma saída analógica de 4-20mA ou 0-10V para indicar o valor da $\%O_2$ da mistura gasosa, sem display gráfico e sem funcionalidades as opcionais, é de 2,049€. A incorporação dum display gráfico tem um custo de 760€. (Preço obtido via email da Lamtec em Abril de 2009)

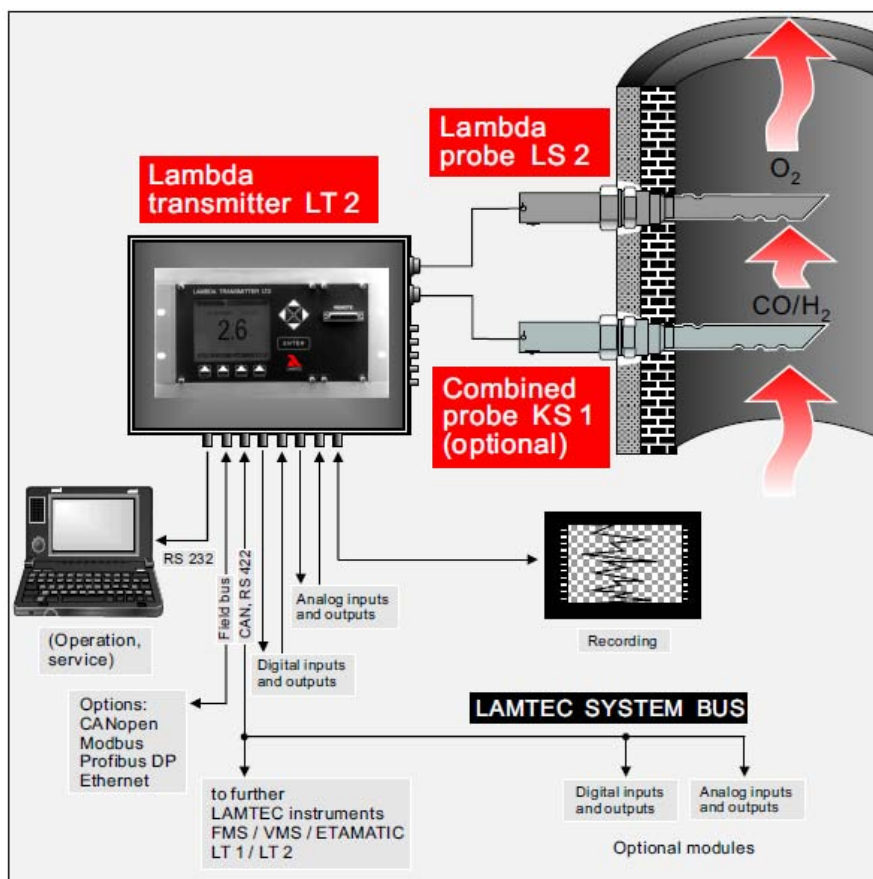


FIGURA 2-4 – INSTRUMENTO DE MEDIDA UNIVERSAL DE PERCENTAGEM DE OXIGÊNIO LT2

O Lambda-Transmitter LT10, apresentado na figura 2.5, é um instrumento de medida universal de concentração de oxigênio, que mede sinais lineares. Controla e monitoriza automaticamente o volume do gás que irá ser mensurado.

As suas principais características são [10]:

- Apresentar elevada precisão, melhor que 0,2% O₂ na gama de 0 a 21% de O₂;
- Apresentar um tempo de resposta inferior a 20 segundos;
- Poder ser utilizado em aplicações cuja temperatura do gás de combustão chegue aos 1400°C;
- Apresentar resultados insensíveis à temperatura;
- Não requerer gases de calibração, calibra automaticamente com o ar atmosférico;
- Requerer baixa manutenção.

O preço deste analisador não foi fornecido pela Lamtec, mas calcula-se que seja da mesma ordem de grandeza dos outros analisadores analisados.

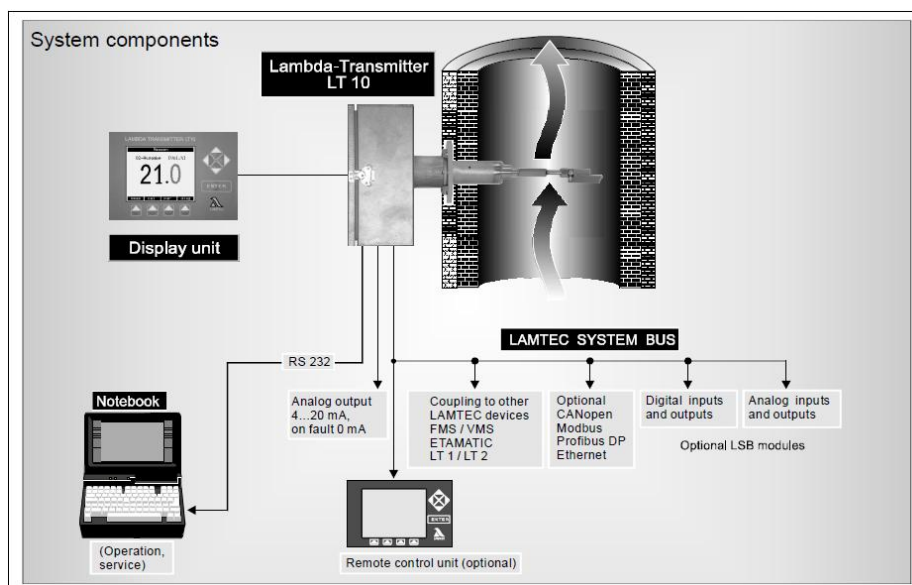


FIGURA 2-5 – INSTRUMENTO DE MEDIDA UNIVERSAL DE PORCENTAGEM DE OXIGÉNIO LT10

Outra empresa, que também produz analisadores baseados na sonda lambda, é a empresa alemã CSM GmbH e tem à disposição mais 5 analisadores de oxigénio baseado na sonda lambda, entre os quais vamos destacar o EGR 5230.

O EGR 5230, apresentado na figura 2.6, é um analisador que mede o teor de oxigénio nos gases de combustão. Possui ainda as funcionalidades, [11] apresentando:

- a %EGR (Exhaust Gás Recirculation),
- o valor de lambda,
- a pressão do gás de combustão.

Tem um tempo de resposta inferior a 1 segundo e uma boa precisão (inferior a 0.4%). O preço deste analisador é de 15,995€ (preço obtido via email da CSM GmbH em Maio de 2009).



FIGURA 2-6 – ANALISADOR DE MEDIÇÃO DOS GASES DE COMBUSTÃO EGR 5230

Em Portugal, não foi encontrada nenhuma empresa que fabrique produtos deste tipo. Porem, sabe-se que a Macolis vende em Portugal caldeiras, com a opcionalidade de controlo da combustão por sonda lambda [12].

2.4. Princípio de funcionamento da Sonda Lambda

A Sonda Lambda é, basicamente, formada por um núcleo cerâmico constituído por dióxido de zircónio (ZrO_2) estabilizado com óxido de ytrio (Y_2O_3) e revestido por camadas de platina porosa [14]. Este núcleo cerâmico possui a forma de dedal, encontrando-se a platina a revestir a sua parte interior e exterior, como se representa na figura 2.6 [13] (note-se que as superfícies de platina interior e exterior não se interligam e constituem os eléctrodos da sonda). A proteger, mecanicamente, este conjunto existe uma carcaça metálica com orifícios, que permitem ao eléctrodo externo estar em contacto com o ar da combustão [14]. A cavidade formada pelo eléctrodo interno está em contacto directo com a atmosfera, que serve de gás de referência.

A camada de platina porosa, que reveste o núcleo, forma uma superfície catalítica, que promove a transformação das moléculas de oxigénio (O_2), moléculas inertes à temperatura ambiente, em iões de oxigénio (O^{2-}).

O dióxido de zircónio estabilizado com óxido de ytrio, quando aquecido a temperaturas superiores a 300°C , apresenta a capacidade de difundir iões de oxigénio (O^{2-}), através dos espaços vazios existentes na estrutura cerâmica em que estão moldados [3].

Dentro do núcleo cerâmico, os iões de oxigénio fluem no sentido da sua maior concentração (normalmente a superfície de platina em contacto com a atmosfera – parte interna da sonda) para a menor (superfície em contacto com os gases de efluentes). Assim, a superfície de platina situada em contacto com o ar atmosférico, possuindo uma percentagem de oxigénio superior aos gases efluentes que contactam com a outra superfície, promove a ionização e fornece os 4 electrões necessários para a formação dos iões O^{2-} , (redução da molécula de oxigénio) ficando com uma carga positiva de menos 4 electrões por cada molécula de oxigénio reduzida. Os iões de oxigénio, cada um com dois electrões a mais, penetram no núcleo cerâmico e difundem-se através deste, de modo a equalizar a sua concentração. Ao chegarem junto da superfície de platina em contacto com uma percentagem de oxigénio inferior, tendem a atravessá-la, fornecendo-lhe 4 electrões, de modo a que cada dois iões se transformem numa molécula de oxigénio (oxidação). Verifica-se, deste modo, o aparecimento de uma diferença de potencial entre os eléctrodos de platina, resultante da superfície interior (pólo positivo e cátodo) possuir falta de electrões e a superfície exterior possuir excesso (pólo negativo e ânodo), comportando-se assim a sonda como uma **Célula Galvânica**, isto é, uma pilha ou bateria.

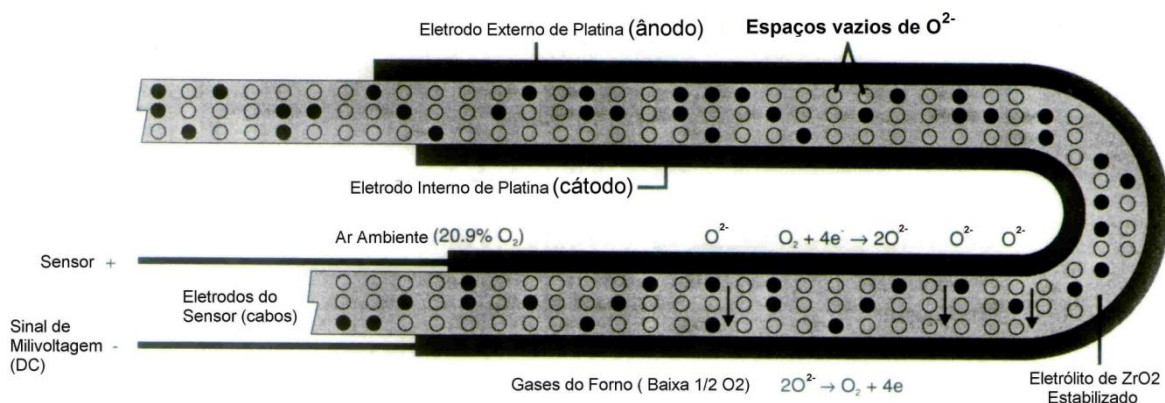


FIGURA 2-7 – PRINCÍPIO DE FUNCIONAMENTO DA SONDA LAMBDA

As semi-reacções de oxidação e redução são, deste modo, acompanhadas por um movimento de electrões, através do núcleo cerâmico, do cátodo para o ânodo, que termina quando se atinge uma determinada diferença potencial entre os dois eléctrodos de platina. As reacções prosseguirão, contudo, quando estabelecermos uma ligação galvânica, por exemplo, através de um milivoltímetro, entre os eléctrodos de platina, pois os electrões em excesso na superfície exterior fluirão através do equipamento de medida para o eléctrodo interior, criando assim uma diferença de potencial, função da diferença de concentrações de oxigénio entre os dois meios.

Os fenómenos acima descritos podem ser matematizados pela Equação de Nernst¹⁴ (2.1) que traduz o modo como a diferença de potencial se relaciona com a diferença de concentração.

$$E = - \frac{R \cdot T}{4 \cdot F} \ln \left(\frac{PO_{2\text{combustão}}}{PO_{2\text{atmosferico}}} \right) \quad (2.1)$$

- E – diferença de potencial gerada entre os eléctrodos da sonda (V);
- R – constante universal dos gases ($8.314472(15) \text{ JK}^{-1}\text{mol}^{-1}$);
- F – Constante de Faraday ($9.64853399(24) \times 10^4 \text{ C mol}^{-1}$);
- T – Temperatura absoluta na sonda (K),
- $PO_{2\text{combustão}}$ – Pressão parcial de oxigénio nos gases de combustão;
- $PO_{2\text{atmosferico}}$ – Pressão parcial de oxigénio na atmosfera.

No caso da pressão nos gases de combustão ser igual à pressão atmosférica, $PO_{2\text{combustão}}$ é o teor de oxigénio nos gases de combustão e $PO_{2\text{atmosferico}}$ é o teor de oxigénio na atmosfera ($\approx 20.95\%$).

¹⁴ Walter Hermann Nernst, físico-químico alemão que ganhou o prémio Nobel de 1920 pela Terceira Lei Termodinâmica.

Como foi atrás referido, este fenómeno só ocorre a temperaturas superiores a 300°C, temperatura a partir do qual o núcleo cerâmico passa a ser condutor de iões O^{2-} . Isto implica que o gás de combustão esteja a uma temperatura superior a 300°C para aquecer o núcleo de dióxido de zircónio estabilizado com óxido de ytrio. Contudo, com o intuito de permitir o funcionamento da sonda, em condições cuja temperatura do gás de combustão seja inferior ao necessário, e com a finalidade de tornar a resposta da sonda praticamente independente da temperatura do gás de combustão, alguns sensores estão munidos de um aquecedor eléctrico, possuindo uma resistência cerâmica que permite manter a temperatura acima dos 350°C. Este elemento cerâmico exhibe características PTC¹⁵, o que permite um aquecimento rápido e reduz a potência que é necessária fornecer ao elemento-aquecedor quando o gás de combustão está quente [5].

Os sensores munidos de um elemento aquecedor adquiriram outras vantagens, face aos sensores originais que não possuíam este elemento, nomeadamente, no que diz respeito à longevidade do sensor (devido a menor deterioração causada pelo stress térmico), à maior flexibilidade do local onde podem ser instalados e ao facto de permitirem uma calibração mais precisa (devido à baixa variação da resposta da sonda com a temperatura). [15]

¹⁵ PTC – Positive Temperature Coefficient, isto é, a resistência aumenta com o aumento da temperatura.

Capítulo 3

3. Visão global da Estrutura do equipamento a desenvolver - especificação e decomposição em blocos funcionais

3.1. Introdução

Este capítulo inicia-se por uma descrição detalhada das especificações do produto a desenvolver. Em seguida, é apresentada a decomposição do mesmo, nos seus principais blocos funcionais, e posteriormente é feita uma breve descrição das tarefas a exercer por cada bloco. Ainda antes de terminar este capítulo, são apresentados os resultados obtidos no primeiro ensaio com a sonda lambda, uma vez que são importantes para se obter uma visão geral sobre o funcionamento da sonda e sobre o *hardware* e *software* a desenvolver.

3.2. Especificações do sistema medidor da percentagem de oxigénio num gás

As especificações do produto a desenvolver, impostas pela Selmatron, foram as seguintes:

- Utilizar, como sensor da percentagem de oxigénio no gás de combustão, uma sonda lambda igual à utilizada no controlo da combustão dos motores dos automóveis;
- Converter o sinal gerado pela sonda lambda num sinal de 4-20mA variando linearmente com a percentagem de oxigénio;
- A saída do medidor deve suportar uma carga máxima de 500 Ω ;
- O Medidor deverá apresentar uma resolução inferior a 0,2% de O₂;
- O erro absoluto não deverá ser superior a 0,5% de O₂.

3.3. Blocos funcionais do sistema

Atendendo ao que já foi apresentado no capítulo anterior, a sonda lambda gera uma tensão, que decresce logaritmicamente com o teor de oxigénio presente numa determinada mistura gasosa (equação 2.1). Um dos requisitos impostos pela Selmatron é que esse sinal seja convertido em 4-20mA e que varie linearmente com o teor de oxigénio dessa mistura. Isto é, pretende-se converter o sinal do gráfico do lado esquerdo no sinal do gráfico do lado direito da figura 3.1.

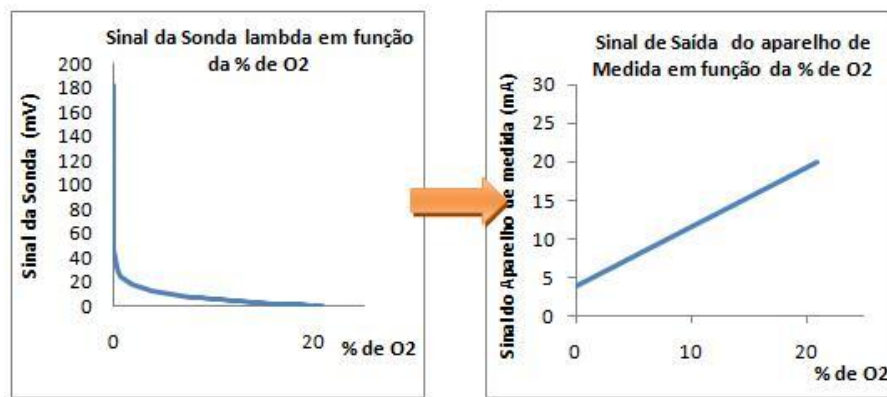


FIGURA 3-1 – CONVERSÃO DUM SINAL LOGARITMICO NUM SINAL LINEAR

Está-se deste modo, perante um problema típico de Instrumentação, que pode ser subdividido em três blocos funcionais: acondicionamento do sinal, aquisição e processamento de dados por um microcontrolador e transmissão dos resultados via loop de corrente activo 4-20mA, conforme ilustrado na figura 3.2.

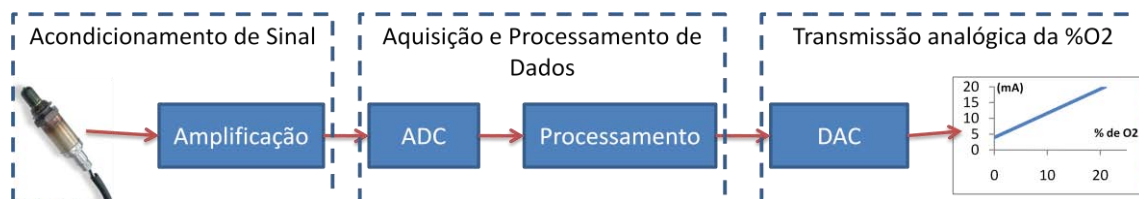


FIGURA 3-2 – BLOCOS FUNCIONAIS DO MEDIDOR A DESENVOLVER

São duas as razões pelas quais se utilizou o protocolo loop de corrente 4-20mA para transmitir a percentagem de oxigénio:

- É um protocolo padrão na indústria, pois grande parte dos dispositivos industriais utiliza este protocolo como forma de comunicação e alimentação;
- Permite detectar o surgimento de um circuito aberto nas linhas de comunicação entre dispositivos, enquanto no protocolo de comunicação 0-20mA, um circuito aberto é confundido como o nível mais baixo do sinal.

3.3.1. Descrição das tarefas a exercer por cada bloco funcional

3.3.1.1. Acondicionamento de sinal:

O sinal proveniente da sonda apresenta-se com amplitudes muito reduzidas e não pode ser directamente utilizado pelo microcontrolador. Cabe a este bloco amplificar o sinal e colocá-lo dentro da gama de conversão do conversor analógico-digital (ADC), isto é, dentro da gama de entrada a que a ADC opera.

3.3.1.2. Aquisição e processamento de dados:

Como o sinal proveniente da sonda varia logaritmicamente com a %O₂ da mistura gasosa que se está a medir, compete a este bloco converter o sinal amplificado pelo bloco anterior numa variação linear função do valor da %O₂, correspondente à mistura que a sonda está a medir. Posteriormente, o valor medido deverá ser codificado correctamente para o código utilizado pelo bloco de transmissão analógica da %O₂.

3.3.1.3. Transmissão analógica do valor de %O₂

É requisito deste problema que o valor da %O₂ seja indicado num sinal analógico entre 4 e 20mA. A função deste bloco será converter um código digital num sinal analógico através de um conversor digital analógico (DAC).

3.4. Determinação do método a utilizar pelo algoritmo a implementar no microcontrolador para calcular a percentagem de oxigénio.

Antes de se iniciar a descrição detalhada do modo como foram implementados os blocos funcionais, a nível de Hardware e Software, considerou-se importante apresentar ainda, neste capítulo, os procedimentos e os resultados obtidos num primeiro ensaio com a sonda lambda, (realizado em Outubro de 2008) e cujas conclusões foram importantes para apurar se o algoritmo a implementar no microcontrolador teria de recorrer a uma tabela ou se poderia basear-se numa expressão matemática, para determinar a %O₂ da mistura.

3.4.1. *Preparação do ensaio*

Antes de se realizar o ensaio com a sonda, foi necessário estudar o modo como esta teria de ser alimentada em termos de tensão de aquecimento da cerâmica condutora de iões O²⁻.

Por [5], verifica-se que o sensor deve ser alimentado, preferencialmente, por uma tensão AC de 12V_{eff}.

Escolheu-se um transformador de 220V/15V possuindo uma potência de 60VA. Este transformador possui duas saídas de 15 V_{rms} (15-0-15), pelo que foi necessário introduzir uma resistência em serie com a sonda, destinada a produzir a queda de tensão necessária, de forma a assegurar que a sonda seja alimentada com 12V_{rms} e uma corrente correspondente à dissipação de 16W na sonda em regime estacionário.

Assim, a resistência a colocar em série terá que ser $R = \frac{(15-12) \cdot V}{P} = \frac{(15-12) \cdot 12}{16} = 2.25\Omega$.

A potência que a resistência terá que suportar será de $P = \frac{(15-12)^2}{R} = 4W$.

Como já foi referido no capítulo 2 (secção 2.4), o elemento que aquece a sonda tem características PTC, isto é, a sua resistência aumenta com a temperatura, o que implica que a corrente necessária em regime transitório seja maior que em regime estacionário, logo, a resistência a colocar em série com a sonda deverá suportar uma potência superior a 4W. Escolheu-se a seguinte combinação de resistências: $1\Omega + (3,3\Omega // 2,2\Omega)$, todas de 5W o que perfaz uma resistência de $2,32\Omega$ que se verificou experimentalmente assegurar os $12V_{rms}$ aos terminais de alimentação da sonda.

3.4.2. Esquema de montagem

Na figura 3.3 apresenta-se o esquema de montagem utilizado nos testes da sonda, assim como a lista do equipamento utilizado.

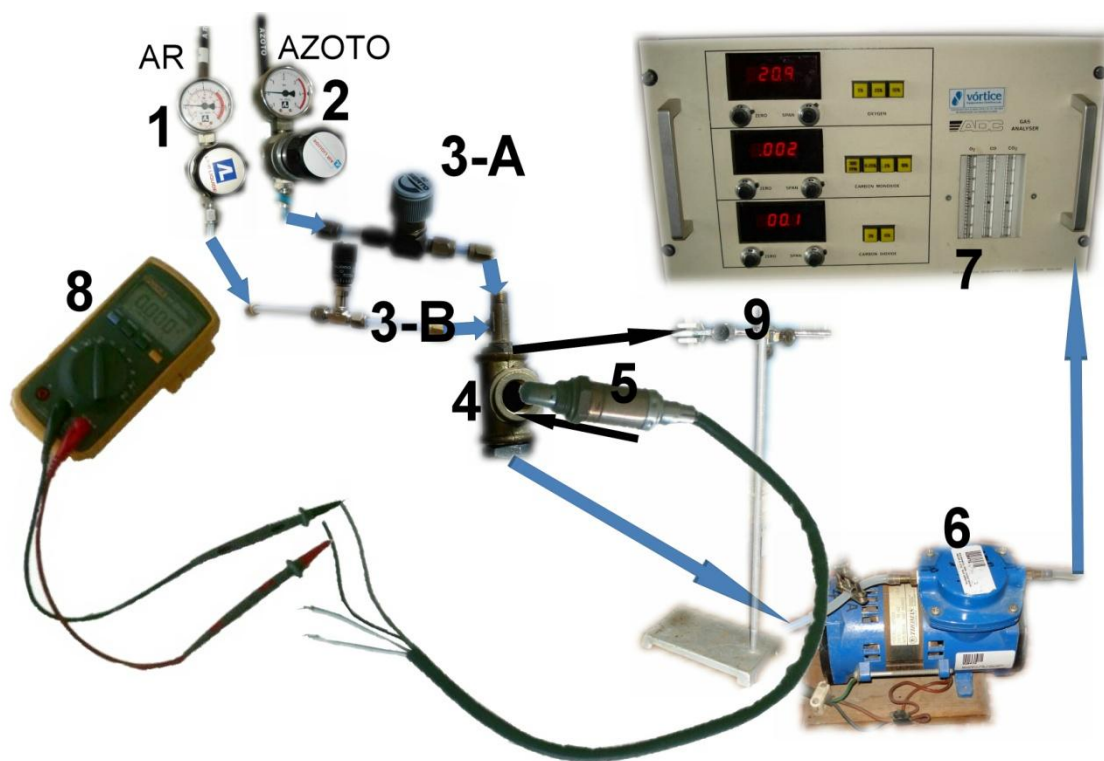


FIGURA 3-3 – FOTOMONTAGEM DO ESQUEMA DE MONTAGEM DO EQUIPAMENTO UTILIZADO NOS ENSAIOS COM A SONDA LAMBDA

- 2 Manómetros redutores de pressão (1 e 2);
- 2 Válvulas de controlo de fluxo de gás (3);
- Tê Latão F/F com encaixe para a sonda (4);
- Sonda de teste (5);
- Bomba de aspiração de gases (6);
- Analisador de Oxigénio (7);
- Multímetro (8);
- Suporte para fixar a sonda (9);
- Tê Latão M/M para interligar as válvulas ao Tê Latão onde encaixa a sonda;
- Tubagem necessária (setas azuis).
- Depósitos de ar e azoto comprimidos

Os equipamentos representados na figura 3.3 foram gentilmente cedidos pelo Departamento de Ambiente e Ordenamento (DAO) da Universidade de Aveiro.

3.4.3. Procedimento de ensaio

- I. Com a válvula A aberta e a válvula B completamente fechada (apenas é fornecido azoto), ajustou-se o potenciómetro “Zero” do Analisador de Oxigénio até aparecer no display o valor 00,1.
- II. Com a válvula A completamente fechada e a válvula B completamente aberta (apenas é fornecido ar), ajustou-se o potenciómetro “Gain” do Analisador de Oxigénio até aparecer no display o valor 20,9 (%O₂ atmosférico – valor que se mantém constante a baixas altitudes).
- III. Repetiram-se os passos I e II, iterativamente até calibrar o Analisador, ou seja, até ele indicar 00,1 no passo I e 20,9 no passo II, sem se tornar necessário ajustar os dois potenciómetros.
- IV. Após a calibração do Analisador actuou-se sobre a válvula A e a válvula B, de modo a alterar a quantidade de azoto e a quantidade de ar presente no interior do T de latão em que está instalada a sonda.
- V. Aguardou-se alguns minutos até a mistura estabilizar e registaram-se os valores recolhidos pelo multímetro e pelo analisador.
- VI. Repetiram-se os passos IV e V com diferentes valores para a mistura azoto-ar, até se obter um conjunto de valores suficientes para traçar o gráfico percentagem de oxigénio – valor do sinal de tensão fornecido pela sonda.

3.4.4. Resultados

Os dados recolhidos foram registados no gráfico 3.1 e verifica-se que, para as misturas ar-azoto ensaiadas, a gama de tensão fornecido pela sonda variava entre -12mV e 170mV.

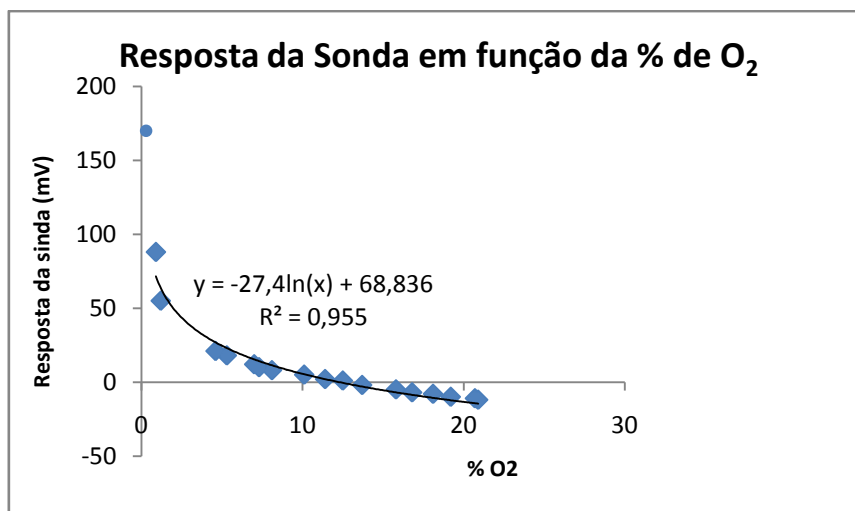


GRÁFICO 3.1 – RESPOSTA DA Sonda EM FUNÇÃO DA PERCENTAGEM DE OXIGÉNIO

3.4.5. Análise dos resultados

A partir do gráfico 3.1, várias conclusões podem ser tiradas. Pode-se concluir que, numa primeira fase, os dados recolhidos permitem modelar a resposta da sonda por uma expressão matemática do tipo logarítmico. Assim, o algoritmo a implementar no microcontrolador, poderá ser um modelo matemático representativo do comportamento da sonda em termos de %O₂-tensão medida.

Repare-se ainda que a curva obtida apresenta a forma logarítmica que, à parte de um erro de offset, corresponde à curva dada pela equação de Nernst já apresentada no capítulo 2 (secção 2.4).

Tomando por base os dados obtidos e utilizando o programa Excel, conseguiu-se, utilizando uma regressão logarítmica, obter a equação que a seguir se apresenta (3.1). Como o coeficiente de correlação $R=0.955$ é próximo da unidade, a equação apresentada descreve, com uma boa aproximação, os dados experimentais, pelo que se tomou a decisão de a utilizar como algoritmo de partida.

$$E = -27,4 \cdot \ln(O_{2_{combustão}}) + 68.836 \Leftrightarrow O_{2_{combustão}} = e^{\left(\frac{E-68,836}{-27,4}\right)}. \quad (3.1)$$

A partir desta equação, pode-se estimar o erro máximo que os valores de %O₂ calculados a partir dessa equação apresentam em relação aos valores lidos directamente do Analisador. O gráfico 3.2 representa, em abcissas, os valores de %O₂ obtidos experimentalmente através do analisador, correspondentes a uma dada

tensão de saída da sonda, e em ordenadas os valores, em %O₂, obtidos, quando se inserem na fórmula matemática, os mesmos valores de tensão. A diferença vertical entre os pontos representados no gráfico e a recta $y=x$ constitui, assim, o desvio entre os valores calculados, a partir da fórmula, e os valores lidos directamente pelo Analisador.

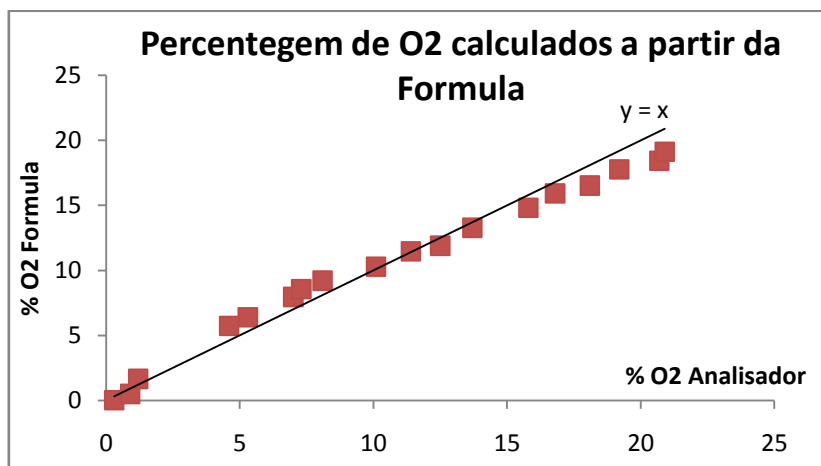


GRÁFICO 3.2 – DISCREPÂNCIA ENTRE OS VALORES OBTIDOS PELO ANALISADOR E OS VALORES CALCULADOS A PARTIR DA EQUAÇÃO 3.1

A partir do gráfico obtido, constata-se que a máxima discrepância entre o valor que o algoritmo gera e o valor que devia gerar, foi registada para valores próximos de 20% de oxigénio com um erro absoluto cerca de 2% de O₂. Embora este valor não cumpra uma das especificações impostas à construção do Medidor (secção 3.2), considerou-se que, nesta fase, os 2% de desvio máximo obtidos seriam aceitáveis, devido à falta de rigor em que os ensaios foram realizados.

3.4.6. Conclusão

Com base nos resultados anteriores, podemos verificar que a determinação do valor de %O₂ da mistura gasosa poderá ser realizada a partir da expressão matemática (3.1) que relaciona a tensão fornecida pela sonda com a %O₂.

Uma vez que o recurso a uma tabela para converter tensão em percentagens de oxigénio implica o recurso a um espaço de memória da PIC relativamente grande, e como se verificou ser possível estabelecer um algoritmo matemático para este efeito, resolveu-se optar por esta última solução.

Capítulo 4

4. Hardware

4.1. Introdução

Neste capítulo é apresentado o hardware desenvolvido para implementar os blocos funcionais que compõem o Medidor. Por razões de clareza de exposição, e contrariamente à ordem seguida pelo autor, durante o desenvolvimento do projecto, os blocos funcionais são apresentados pela seguinte ordem: Bloco de Transmissão Analógica do valor de %O₂; Bloco de Aquisição e Processamento de Resultados e, por fim, o Bloco de Acondicionamento de Sinal.

Como o requisito da resolução da %O₂ tem influência directa sobre o hardware, considera-se importante iniciar este capítulo pela determinação do número de bits necessários para o conversor analógico-digital e digital-analógico.

4.2. Determinação do número de bits necessário para o conversor analógico-digital

Conforme os resultados recolhidos na secção 3.4 do capítulo anterior, o sensor apresenta uma resposta que decresce logaritmicamente de acordo com a expressão $E(O_2) = -27,4 \cdot \ln(O_{2_{combustão}}) + 68.836$.

Dado que esta expressão matemática é uma função contínua estritamente decrescente, a menor variação de tensão da sonda para se obter uma resolução de %O₂ inferior a 0,2%, ocorre na transição do penúltimo nível de %O₂ para o ultimo nível de %O₂ e é dada por $E(20,8) - E(21,0) = 0,26\text{mV}$.

Assim, o número de níveis terá de ser superior a $N^{\circ} \text{ de níveis} \geq \frac{\text{Gama de tensão de saída do sensor}}{\text{menor variação da sonda para uma resolução de } 0,2\% \text{ } O_2} = \frac{170 - (-12)}{0,26} = 700$, o que implica um número de bits mínimo de 10. ($n > \log_2(700 + 1) \approx 9.45$)

4.3. Determinação do número de bits necessário para o conversor digital-analógico

O número de níveis mínimo para o conversor digital analógico é de N° de níveis $\geq \frac{\text{Gama de percentagem de Oxigenio}}{\text{resolução de } O_2 \text{ requerida}} = \frac{20,9}{0,2} = 105$, o que implica um número de bits mínimo de 7. ($n > \log_2(105 + 1) \approx 6.73$)

4.4. Bloco de Transmissão Analógica do valor de %O₂

Para implementar o Bloco de Transmissão Analógica do valor de %O₂, recorreu-se a um *Chip* da Analog Device – AD420.

O AD420 é um conversor Digital-Analógico, implementado num único *Chip*, bastante aplicado no controlo industrial para gerar *outputs* analógicos de 4-20mA, 0-20mA ou 0-24mA.

Apresentam-se a seguir as suas principais características [16]:

- Chip único que gera um sinal de saída entre 4–20mA, 0–20mA ou 0–24mA;
- Tensão de alimentação compreendida entre 12-32V.
- Output Voltage Compliance¹⁶ de 0V a VDD – 2.5V
- Interface Digital Série flexível (3.3 MBPS) que suporta o modo *Tree-Wire* ou o modo assíncrono.
- 16 Bits de resolução e tem comportamento Monotónico¹⁷
- Erro de Não-Linearidade Integral Máximo de $\pm 0.012\%$
- Offset Máximo $\pm 0.05\%$ (ajustável)
- Erro Total de Saída Máximo 0.15% (ajustável)

¹⁶ Output Voltage Compliance – tensão máxima que o integrado suporta no pino *lout* quando a corrente de saída corresponde ao valor para o qual foi programado.

¹⁷ Diz-se que uma DAC não tem um comportamento monotónico quando a tensão de saída não aumenta (sempre) quando o valor da palavra digital de entrada aumenta, ou não diminui (sempre) quando o valor digital de entrada diminui.

FUNCTIONAL BLOCK DIAGRAM

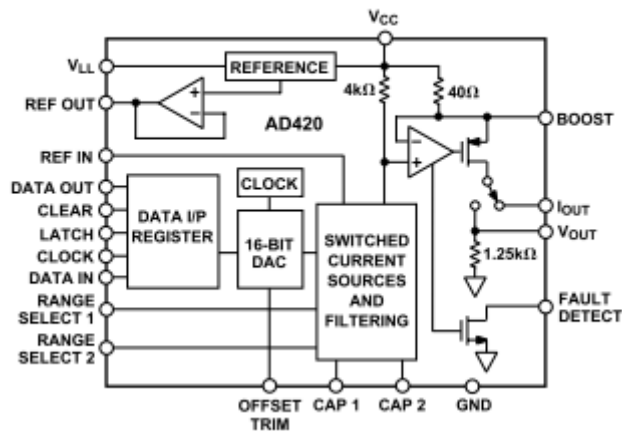


FIGURA 4-1 – AD420

Na figura 4.1 está apresentado o esquema eléctrico base do AD420 [16] e na 4.2 o seu “pin-out”.

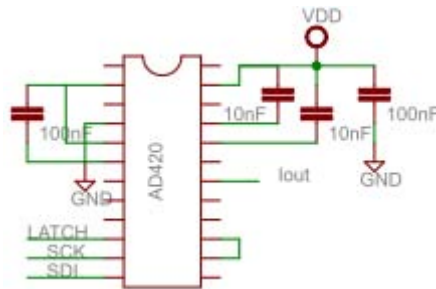


FIGURA 4-2 – ESQUEMA ELÉCTRICO BASE DO AD420

O custo deste integrado ronda os 15,98€ [17].

4.5. Bloco de aquisição e processamento de dados

Nos parágrafos que se seguem, é apresentada a solução encontrada para desenvolver o bloco de Aquisição e Processamento de Dados, responsável pela linearização do sinal gerado pela sonda, ou seja, responsável por converter um sinal que varia logaritmicamente com a %O₂ num sinal que varie linearmente com a mesma.

Este bloco deve ser implementado através de um microcontrolador que possua as seguintes características:

- Elevada capacidade de processamento para concretizar operações exponenciais, multiplicação de inteiros de 16bits e a divisão de inteiros de 32bits por 16bits;
- Comunicação série por SPI para poder comunicar com o AD420, e que este módulo possua preferencialmente um Shift-Register de 16bits;
- ADC com 10bits de resolução no mínimo.

O baixo custo dos microcontroladores da Microchip e a familiarização com este tipo de microcontroladores ao longo do curso, bem como com as ferramentas de compilação de código e de programação dos microcontroladores, determinaram que seria recomendável a escolha de um microcontrolador deste fabricante.

Os microcontroladores da família PIC24H, pelas suas características, apresentam-se como uma boa opção para desenvolver este bloco. Seguidamente são apresentadas as principais características destes microcontroladores, que o tornam uma boa opção para esta aplicação [18]:

- Arquitectura de 16 bits: suporta a multiplicação de 16 x 16 bits e a divisão de 32/16bits e 16/16bits.
- 12K bytes de Memória *Flash*
- 1024 bytes Data SRAM
- Comunicação serie, 4-Wire SPI que suporta tanto 8-bit e 16-bit de dados. Suporta todos os modos de operação no que diz respeito à fase e polaridade do sinal SCK (*serial clock*).
- ADC de 10-bit, 1.1Msps ou 12-bit, 500Ksps, com um Erro de Não- Linearidade Integral máximo de $\pm 2\text{LSB}$ e com um Erro de Não-Linearidades Diferencial máximo de $\pm 1\text{LSB}$. A conversão pode ser iniciada manualmente ou automaticamente sincronizada com uma das 4 fontes de *trigger*.
- *Timer/Counters* de 16 bits com possibilidade de concatenação de dois *timers* para formar um único de 32bits, o que é útil para construir temporizadores da ordem dos minutos.
- Conjunto de instruções optimizadas para o compilador C

O custo deste integrado ronda os 2,12€ [17].

Na figura 4.3 está apresentado o “pinout” base do PIC24HJ12GP201.

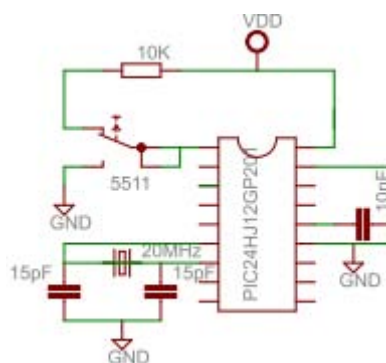


FIGURA 4-3 – ESQUEMA ELÉCTRICO BASE DO PIC24HJ12GP201

4.6. Bloco de acondicionamento de sinal

Conforme o que já foi divulgado na secção 3.4.5 do capítulo 3, a sonda fornece um sinal compreendido entre -15mV e 170mV. Ora, a ADC incorporada no microcontrolador seleccionado na secção anterior, não permite a conversão de sinais com amplitude negativa. Além disso, a ADC converte um sinal de amplitude positiva compreendido entre V_{ref-} e V_{ref+} com uma resolução de 10 ou 12bits. V_{ref-} e V_{ref+} devem cumprir a seguinte condição $0 \leq V_{ref-} < V_{ref+} \leq 3.3V$. Deste modo, o sinal proveniente da sonda não poderá ser directamente utilizado pela ADC, será necessário pegar nesse sinal, amplificá-lo e acrescentar-lhe um offset, para poder ser utilizado pela ADC.

Nos próximos parágrafos são apresentadas várias soluções desenvolvidas para implementar este bloco.

4.6.1. Primeira solução – Amplificação + Offset.

A primeira solução desenvolvida consistiu em projectar um circuito, que convertesse qualquer sinal da gama [-12, 170] mV num sinal da gama [0, 3300] mV, ou seja, um circuito que pudesse ser descrito pela equação:

$$y = G \cdot x + \text{Off}, \quad (4.1)$$

Em que $G = \frac{3300 - 0}{170 - (-12)} = 18.13$ e $\text{Off} = -18.13 \cdot (-12) = 217.58\text{mV}$.

O circuito representado na figura 4.4 ilustra essa solução.

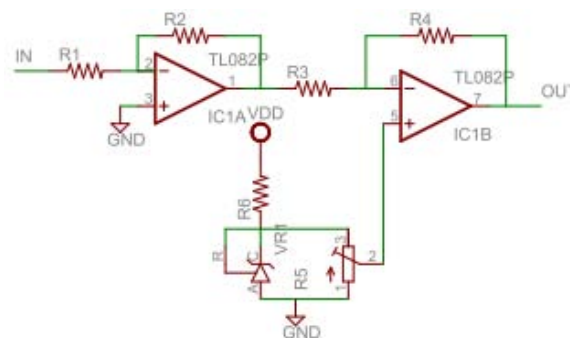


FIGURA 4-4 – ESQUEMA ELÉCTRICO DA PRIMEIRA SOLUÇÃO PARA O CIRCUITO DE ACONDECIONAMENTO DE SINAL

Este circuito é constituído por dois op-amps em configuração inversora. É de relembrar que a resistência de saída da sonda (R_s) é de 250 Ω e está em série com R_1 . Para efeitos da análise que se segue, vai-se considerar $R_{1eq} = R_1 + R_s$.

Análise do circuito:

$$\frac{V_6 - V_7}{R_4} = \frac{V_1 - V_6}{R_3} \Leftrightarrow V_7 = -\frac{R_4}{R_3} V_1 + \left(1 + \frac{R_4}{R_3}\right) V_6$$

$$V_7 = \frac{R_2 \cdot R_4}{R_{1eq} \cdot R_3} V_{IN} + \left(1 + \frac{R_4}{R_3}\right) V_5. \quad (4.2)$$

O ultimo andar de ganho foi projectado para ter um ganho $G_2 = -1$, o que equivale a dizer que $R_3 = R_4$, logo a equação reduz-se a, $V_{OUT} = \frac{R_2}{R_{1eq}} V_{IN} + 2V_5$. Fazendo uma analogia com a equação 4.1 vem que, $R_2 = 18.13 R_{1eq}$ e $V_5 = 108.79 \text{ mV}$. Arbitrando $R_1 = 1 \text{ k}\Omega$, R_2 será de $18 \text{ k}\Omega$.

Para gerar a tensão V_5 , utiliza-se o circuito integrado TL431 (integrado da parte inferior da figura 4.4) que garante uma tensão de 2.5 V no terminal Ref, quando polarizado com uma corrente da ordem dos 10 mA [19]. Em paralelo com o TL431, coloca-se um potenciómetro multi-volta de 10 K que permite gerar a tensão V_5 .

$$R_6 = \frac{V_{DD} - 2.5}{10 \text{ mA}}.$$

4.6.1.1. Testes e resultados

Testes entretanto realizados, demonstram que a sonda apresenta um certo grau de não reprodutividade¹⁸, em resultado das variações de pressão atmosférica e pressão parcial do gás a medir.

O gráfico 4.1 apresenta os resultados recolhidos em Outubro de 2008 (pontos experimentais a vermelho) e os resultados recolhidos a 18 de Fevereiro de 2009 (pontos experimentais a azul).

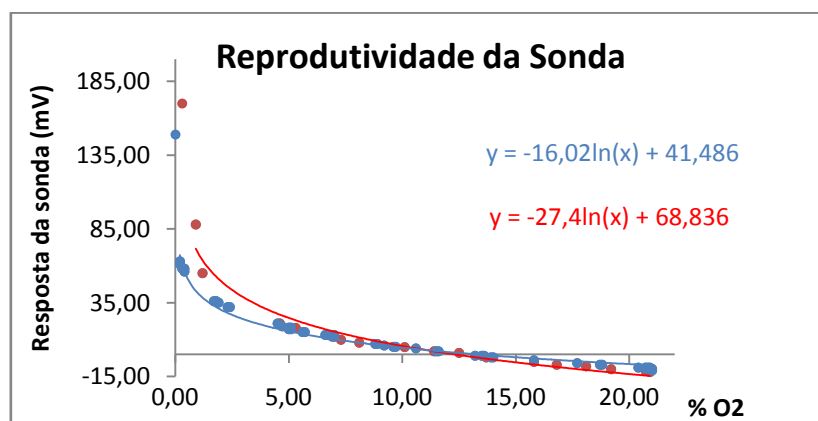


GRÁFICO 4.1 – AVALIAÇÃO DA REPRODUTIVIDADE DA Sonda LAMBDA

¹⁸ Uma resposta diz-se reproduzível se o resultado entre sucessivas medições for insensível às condições em que é exposto.

Por análise ao gráfico, verifica-se que a resposta se mantém praticamente reproduzível a partir dos 2% de Oxigénio. Todavia, as equações das curvas, que melhor traduzem os resultados experimentais, são diferentes.

Perante estes factos, este circuito não se revela uma boa solução, por não tomar em consideração as variações da resposta da sonda, face a novas condições ambientais.

Uma possível solução, para que este circuito possa acomodar as alterações ambientais, é a calibração por software sempre que estas se alterem, o que exige, por parte do técnico ou utilizador deste produto, conhecimentos mínimos de programação para reprogramar os microcontroladores com os novos parâmetros e exige disponibilidade para recolher novos dados, já para não falar das condições necessárias para realizar a recolha de dados.

Estas exigências não agradam aos potenciais clientes, pois o mercado oferece-lhes soluções mais “*User-friendly*”. Esta é a principal razão pela qual se rejeitou esta solução.

4.6.2. Segunda solução – Amplificação variável + offset variável

Antes de apresentar a solução alternativa que permite acomodar facilmente a não reprodutibilidade da sonda, é necessário fazer uma análise detalhada do sinal, à saída do circuito da figura 4.4 da primeira solução em função da %O₂.

4.6.2.1. Análise do sinal à saída do circuito da figura 4.4 em função da %O₂

Repare-se que a tensão à saída do circuito anterior, figura 4.4, (tensão à entrada da ADC) pode ser expressa em função da %O₂ de acordo com a equação:

$V_{ADC}(\%O_2) = V_{OUT}(\%O_2) = G \cdot V_{IN}(\%O_2) + Off$, onde G é o ganho e Off é o offset

Como, $V_{IN}(\%O_2) = m \cdot \ln(\%O_2) + b$, onde b é ordenada na origem e m é o declive, vem que:

$$V_{ADC}(\%O_2) = G \cdot [m \cdot \ln(\%O_2) + b] + Off,$$

$$V_{ADC}(\%O_2) = m \cdot G \cdot \ln(\%O_2) + b \cdot G + Off.$$

Considerando $M = m \cdot G$ e $B = b \cdot G + Off$ obtém-se:

$$V_{ADC}(\%O_2) = M \cdot \ln(\%O_2) + B. \quad (4.3)$$

Esta equação (4.3) é importante para corrigir a não reprodutibilidade da sonda. A ideia desta segunda solução consiste em programar uma vez o microcontrolador com os parâmetros M e B. Sempre que se verificar que a sonda se encontra descalibrada, o técnico ou o utilizador poderá proceder à sua calibração por reajustes dos potenciômetros de ganho e offset. O mesmo equivale a dizer que sempre que, os parâmetros da sonda, \underline{m} e \underline{b} , se alterarem em consequência das alterações ambientais (pressão, humidade, outros), os valores dos parâmetros M e B podem ser recuperados por ajuste dos potenciômetros de ganho e offset, G e off.

4.6.2.2. Método de calibração

O método de calibração consiste em expor a sonda a dois gases padrão de calibração cuja %O₂ seja conhecida e ajustar os potenciômetros, até que o valor da %O₂ indicada no monitor seja igual à %O₂ do gás padrão. Este é um processo iterativo que só termina quando o microcontrolador indicar correctamente a %O₂ de cada gás, sem ser necessário ajustar os potenciômetros.

4.6.2.3. Escolha dos valores dos parâmetros que traduzem a resposta da sonda à entrada da ADC, M e B

Verificando-se que, na prática, misturas gasosas com %O₂ inferior a 0,4%, raramente aparecem, decidiu-se projectar os parâmetros M e B, de forma a que o limite inferior do Medidor fosse 0,4% de O₂, o que permite melhorar significativamente a resolução do Medidor de %O₂, uma vez que para valores abaixo deste a tensão fornecida pela sonda sobe abruptamente.

Como a ADC devolve um valor nulo quando a tensão de entrada é negativa, o que pode ocorrer em situações de calibração do equipamento, tornou-se necessário assegurar que a tensão à entrada da ADC correspondente a 21% de O₂ seja não negativa e não nula, isto para não se correr o risco, durante o processo de calibração, de o microcontrolador estar a devolver ao utilizador a indicação que a %O₂ é de 21% quando a tensão da ADC é negativa. O que pode causar calibrações indesejadas.

Assim, os parâmetros M e B foram projectados de modo a que ao Nível 4095 correspondesse a 0,4% de O₂ e ao nível 0 correspondesse a 22% de O₂, ou seja, $V_{ADC}(0,4\%)=2.5V$ e $V_{ADC}(22.0\%)=0V$. Deste modo, o microcontrolador devolve um valor compreendido entre 0,4 e 22%, o que permite ao técnico/utilizador ter percepção se a sonda se encontra calibrada quando o gás a medir for o ar atmosférico (21% de O₂), pois nesta situação o microcontrolador deve indicar 21%.

$$\begin{cases} 4095 = M \cdot \ln(4) + B \\ 0 = M \cdot \ln(220) + B \end{cases} \Leftrightarrow \begin{cases} 4095 = M \cdot \ln(4) + (-M \cdot \ln(220)) \\ B = -M \cdot \ln(220) \end{cases} \Leftrightarrow$$

$$\begin{cases} 4095 = M \cdot \ln(4/220) \\ - \end{cases} \Leftrightarrow \begin{cases} M = \frac{4095}{\ln(4/220)} = -1021.9 \\ B = 5511.6 \end{cases} \quad (\text{os valores de \%O}_2 \text{ foram}$$

multiplicados por 10 de modo a incluir o valor decimal como parte inteira.

4.6.2.4. Circuito de amplificação variável + offset variável

O circuito da figura 4.5 apresenta o hardware desenvolvido para implementar a segunda solução.

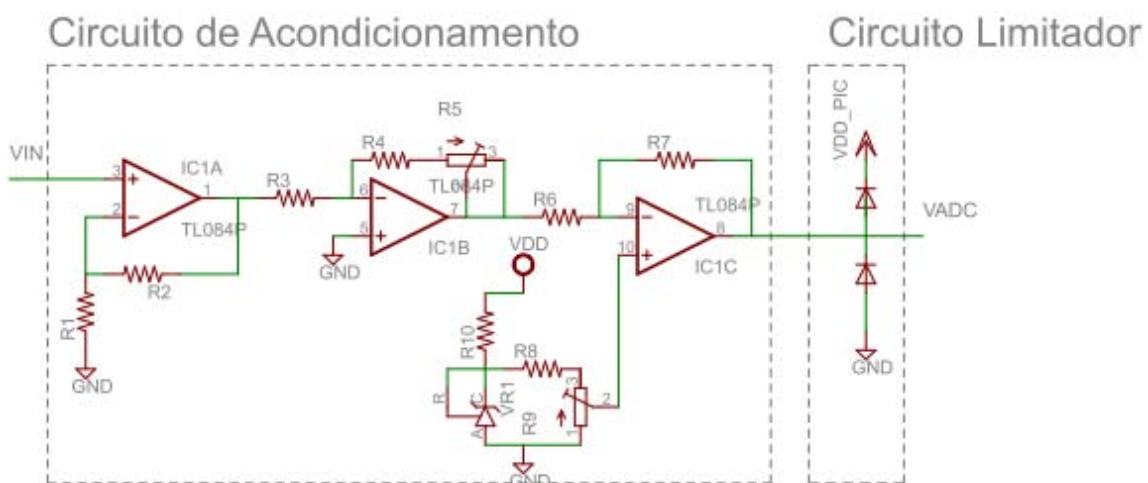


FIGURA 4-5 – ESQUEMA ELÉCTRICO DA SEGUNDA SOLUÇÃO PARA O CIRCUITO DE ACONDICIONAMENTO DE SINAL

O circuito foi projectado para ter 3 andares de ganho. O primeiro andar de amplificação tem ganho próximo de 11, pois $R1=1K\Omega$ e $R2=10K\Omega$, e foi montado em configuração não inversora, por esta configuração apresentar uma impedância infinita perante a sonda e para não carregar os andares seguintes. O segundo andar foi montado em configuração não inversora e tem ganho variável. Por fim, o último andar tem ganho unitário ($R7=R6=1K\Omega$), o qual permite acrescentar o offset.

À saída do circuito foram acrescentados 2 díodos de *Schottky*, cujo objectivo é proteger a entrada da ADC contra tensões acima/abaixo do limite superior/inferior que ela suporta, uma vez que ela poderá ficar exposta a essas condições durante o processo de calibração, quando a $\%O_2$ for inferior a 0,4% ou quando alguém desajustar os potenciômetros.

O circuito foi projectado para utilizar a tensão gerada pelo TL431 como V_{ref+} da ADC, pois esta revela-se mais estável que a tensão de alimentação do microcontrolador.

Foi utilizado um TL084 para implementar os 3 andares de ganho. Este circuito possui um elevado factor de rejeição de modo comum (80dB), uma elevada impedância de entrada (1 TΩ) e um elevado factor de rejeição de tensão de alimentação (80dB) [20]. Este circuito tem um custo de 0,21€ [17].

O custo de um TL431 ronda os 0,13€ [17].

4.6.2.5. Determinação da gama de variação do ganho

Com o intuito de determinar o ganho mínimo e o ganho máximo que se deve aplicar ao segundo andar de amplificação, mergulhou-se a sonda em misturas gasosas, com diferentes concentrações de oxigénio, e recolheu-se os valores de tensão à saída do primeiro op-amp, V1, os quais estão apresentados no gráfico 4.2.

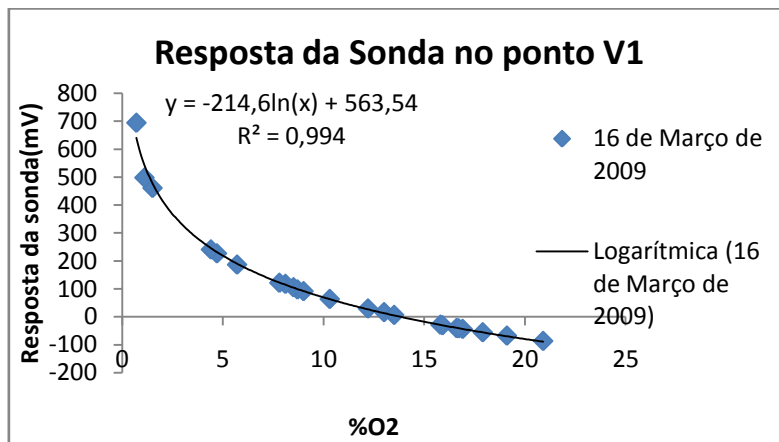


GRÁFICO 4.2 – RESPOSTA DA Sonda À SAÍDA DO PRIMEIRO ANDAR DE GANHO

Estipulou-se que o desvio máximo que a sonda poderia sofrer em relação a estes valores seria de 20%. Conforme a equação, que representa a resposta da sonda no ponto V1, o ganho máximo é dado por:

$$G2_{\max} = \frac{2500}{0.8(-214.6\ln(0.4)+563.54)-0.8(-214.6\ln(22)+563.54)} = 3.6 \quad (4.4)$$

O ganho mínimo é dado por:

$$G2_{\min} = \frac{2500}{1.2(-214.6\ln(0.4)+563.54)-1.2(-214.6\ln(22)+563.54)} = 2.4. \quad (4.5)$$

Pela análise do circuito, o ganho G2 é dado por $G = -\frac{R4+R5p}{R3}$. (4.6)

Por 4.4 e 4.5 e 4.6,

$$\begin{cases} 3.6 = \left| -\frac{R4 + R5p}{R3} \right| \\ 2.4 = \left| -\frac{R4}{R3} \right| \end{cases} = \begin{cases} 3.6R3 = R4 + R5p \\ 2.4R3 = R4 \end{cases} = \begin{cases} 3.6R3 = 2.4R3 + R5p \\ 2.4R3 = R4 \end{cases}$$

$$= \begin{cases} R5p = 1.2R3 \\ R4 = 2.4R3 \end{cases}$$

Utilizou-se um potenciómetro multi-volta de 2K $\Rightarrow R3=1.66K\Omega \approx 1K5\Omega$ e $R4=3K6\Omega \approx 3K9\Omega$.

A tensão V9 é gerada do mesmo modo que a tensão V5 do circuito apresentado na primeira solução (secção 4.6.1).

4.7. Alimentação dos diferentes blocos funcionais.

Por [12], a tensão de alimentação da DAC deve estar 2,5V acima da tensão máxima do pino de saída, ou seja,

$$V_{DD} > V_{out_{DAC}} + 2.5V = 20mA \cdot 500\Omega + 2.5V = 12.5V.$$

A tabela 4.1 apresenta as tensões com que devem ser alimentados os integrados utilizados no projecto.

| | TL084 | | | PIC24HJ12GP201 | | | AD420 | | |
|------------------------|-------|-----|-----|----------------|-----|------|-------|-----|-----|
| | MIN | TYP | MAX | MIN | TYP | MAX | MIN | TYP | MAX |
| Tensões de alimentação | - | - | 18V | 3.0V | - | 3.6V | 4.5V | - | 36V |

TABELA 4.1 – ALIMENTAÇÕES DOS INTEGRADOS CONSTITUINTES DO SISTEMA

A figura 4.6 representa o esquema que fornece as tensões de alimentação para os diferentes blocos.

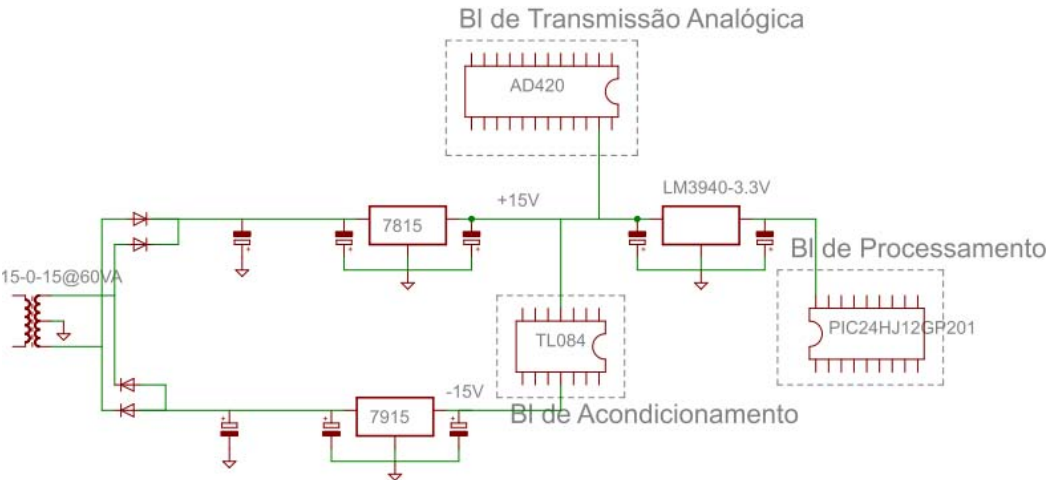


FIGURA 4-6 – ESQUEMA DE ALIMENTAÇÃO DOS DIFERENTES BLOCOS DO MEDIDOR

4.8. Interligação entre os diferentes blocos

Nesta secção é apresentada a forma como os diferentes blocos estão interligados. É feita uma breve descrição do protocolo *Serial Peripheral Interface* (SPI), uma vez que este protocolo é usado para comunicar entre a DAC e o PIC24HJ.

4.8.1. Interligação entre o bloco de acondicionamento e o bloco de aquisição e processamento de dados

Como se pretende utilizar uma fonte de tensão de referência externa, para realizar a conversão analógico-digital, a entrada analógica AN0 fica automaticamente impossibilitada de ser utilizada como um canal de entrada da ADC, pois esta entrada partilha o mesmo pino de entrada com o canal Vref+ da ADC. De entre as restantes 5 entradas analógicas que este PIC possui (AN1...AN3, AN6 e AN7), decidiu-se ligar o sinal de saída do Bloco de Acondicionamento de Sinal à entrada AN1 (pino 3) conforme indicado na figura 4.7.

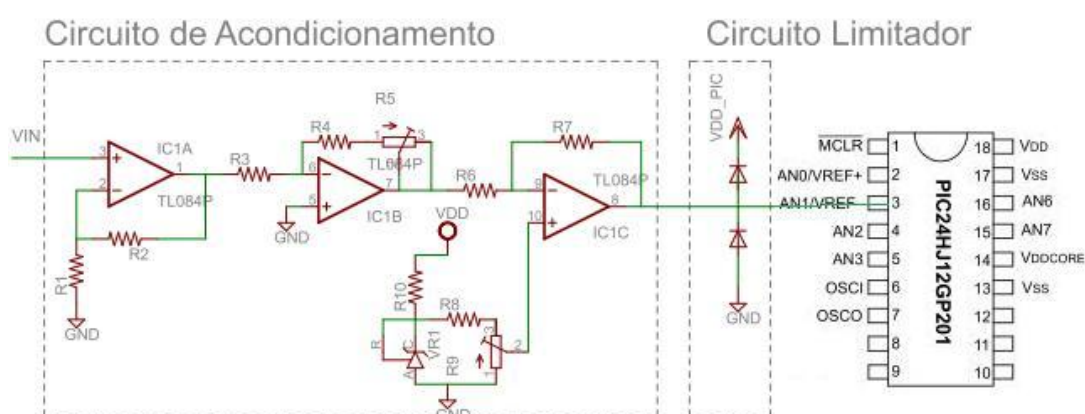


FIGURA 4-7 – ESQUEMA ELÉCTRICO DE LIGAÇÃO DO CIRCUITO DE ACONDICIONAMENTO AO PIC24HJ12GP201

4.8.2. Interligação entre bloco de aquisição e processamento de dados e o bloco de transmissão analógica da %O₂

A comunicação entre o PIC24HJ e a DAC é feita por *Serial Peripheral Interface* (SPI). Como tal, nos parágrafos que se seguem, descrever-se-á este protocolo de comunicação.

4.8.2.1. Serial Peripheral Interface

O barramento SPI é um *link* de comunicação de dados síncrona, implementado inicialmente pela Motorola. A comunicação entre os dispositivos segue a arquitectura *Master/Slave* com ligação ponto a ponto, sendo o *Master* o responsável por iniciar sempre a comunicação. O protocolo admite vários *Slaves* desde que exista uma linha de *Slave Select* para cada *Slave* [21].

O protocolo define 4 sinais distintos para realizar a comunicação:

- SCK – Serial Clock;
- SDI – Serial Data In;
- SDO – Serial Data Out;
- SS – Slave Select.

O pino SDI do *Master* é ligado ao pino SDO do *Slave*, e vice-versa. Normalmente, numa transmissão de dados são utilizados dois *shift register*, um no *Master* e outro no *Slave* conectados em forma de anel [21].

Antes de iniciar a comunicação entre o *Master* e o(s) *Slave(s)*, deve-se configurar correctamente o sinal *clock* do *Master* com uma frequência menor ou igual à frequência máxima que o(s) *Slaves* suportam. Deve-se configurar também o modo de validação de dados, configurando a polaridade (CPOL) e a fase (CPHA) da transmissão de acordo com as especificações impostas pelos *Slaves* [21] (tabela 4.2).

| CPOL | CPHA | Modo de Validação de dados |
|------|------|---|
| 0 | 0 | Os dados são lidos na transição ascendente do relógio e são alterados na transição descendente. O estado <i>Idle</i> do SCK é o nível 0 e o estado <i>active</i> é o nível 1. |
| 0 | 1 | Os dados são lidos na transição descendente do relógio e são alterados na transição ascendente. O estado <i>Idle</i> do SCK é o nível 0 e o estado <i>active</i> é o nível 1. |
| 1 | 0 | Os dados são lidos na transição descendente do relógio e são alterados na transição ascendente. O estado <i>Idle</i> do SCK é o nível 1 e o estado <i>active</i> é o nível 0. |
| 1 | 1 | Os dados são lidos na transição ascendente do relógio e são alterados na transição descendente. O estado <i>Idle</i> do SCK é o nível 1 e o estado <i>active</i> é o nível 0. |

TABELA 4.2 – TABELA COM OS DIFERENTES MODOS DE VALIDAÇÃO DE DADOS: POLARIDADE E FASE

Após a configuração do *clock*, da polaridade e da fase, normalmente, o *Master* coloca o sinal SS do *Slave* com que pretende comunicar com o nível lógico baixo. Se for exigido um período de espera (como acontece nas conversões analógica-digital), o *Master* terá de aguardar até esse período terminar, antes de iniciar a transmissão de dados [21].

Ainda antes de iniciar a transmissão, os *shift registers* são carregados com os dados a transmitir. Posteriormente, seguem-se um número fixo de ciclos de relógio igual ao tamanho dos *shift registers*.

Durante cada ciclo de relógio SPI ocorre uma transmissão de dados *full duplex*, ou seja, o *Master* envia um bit pela linha SDO que será lida pelo *Slave*, ao mesmo tempo o *Slave* envia um bit pela linha SDO (SDI do lado *Master*) que será lido pelo *Master*.

Enquanto houver dados para transmitir, os *shift registers* são carregados com os dados a transmitir. Quando não houver mais dados para trocar, o *Master* deixa de emitir ciclos de *clock* e coloca um nível lógico alto na linha SS.

4.8.2.2. Aplicação do protocolo SPI entre o PIC24HJ e a AD420

Antes de mais, foi necessário identificar o *Master* e o *Slave*. Indiscutivelmente, o *Master* é o microcontrolador e o *Slave* é a DAC. Assim, por consulta ao *datasheet* do AD420, verifica-se que a frequência máxima de operação deste dispositivo é de 3,3MHz. O SCK foi programado para 1,25MHz.

A figura 4.8 [16] permite-nos identificar a fase e a polaridade dos dados.

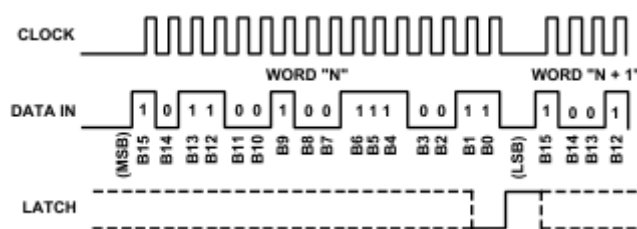


FIGURA 4-8 – DIAGRAMA DE TEMPO DOS SINAIS SPI IMPOSTA PELA AD420

Observando a figura 4.8, verifica-se que o estado *IDLE* do *CLOCK* é o nível 0 e que a alteração de dados é efectuada na transição descendente. Portanto CPOL = 0 e CPHA = 1.

Para gerar o sinal *LATCH* não se utilizou o SS do PIC, este sinal é controlado por *software* que aguarda 17 ciclos de relógios SCK para colocar um pino de saída (o pino 12) no estado *high*.

Foi ainda necessário configurar os pinos que deveriam ser usados para os sinais SCK e SDI, uma vez que os microcontroladores da família PIC24H não têm pinos fixos para cada periférico, estes podem ser re-mapeados. Esta propriedade confere vantagens aos projectistas, uma vez que deixam de existir conflitos entre periféricos que, noutros microcontroladores, partilham o mesmo pino de entrada/saída. Deste modo, os periféricos deixam de estar limitados pelos pinos de I/O.

O microcontrolador foi programado para usar o pino 10 como SDI e o pino 11 como SCK, conforme ilustrado na figura 4.9.

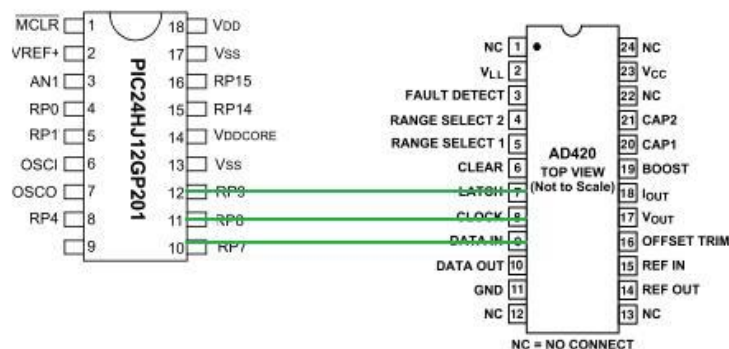


FIGURA 4-9 – ESQUEMA ELÉCTRICO DE LIGAÇÃO PIC24HJ12GP201 – AD420

4.8.3. Interligação microcontrolador – PC

Para os utilizadores que não estão interessados em saber a $\%O_2$ em termos de corrente, mas que pretendem ler directamente o valor da $\%O_2$, projectou-se o circuito para comunicar com um PC via RS-232. Foi necessário configurar os pinos associados à UART, pela mesma razão que já foi referida para o SPI. Assim, o microcontrolador foi programado para usar o pino 15 como RX da UART e o pino 16 como TX da UART.

A pedido da Selmatron, o circuito final deveria possuir o menor número de componentes possível para minimizar as dimensões da placa do Medidor de $\%O_2$.

Mediante este facto, decidiu-se utilizar a MAX232 externamente à placa do Medidor de $\%O_2$, mais concretamente no cabo de comunicação com o PC, uma vez que esta só é efectivamente necessária, quando o PC estiver conectado ao microcontrolador, poupando-se, deste modo, também o consumo da MAX232, quando ela não é efectivamente necessária.

Deste modo, os pinos RX e TX entram directamente num conector do tipo RJ45 que é mais robusto que o conector RS-232, e portanto, para comunicar com o PC será necessário um cabo RJ45 – RS-232 com MAX232 incorporada e um cabo com conversor RS232-USB, no caso de o PC não possuir uma interface para RS-232.

A figura 4.10 apresenta o esquema de ligação entre o PIC e o conector RJ45.

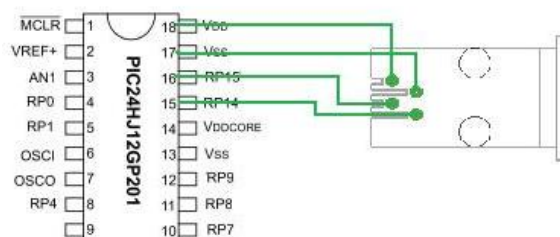


FIGURA 4-10 – ESQUEMA ELÉCTRICO DE LIGAÇÃO PIC – CONECTOR RJ45

A figura 4.11 apresenta o esquema eléctrico de ligação do cabo RJ45 ao MAX232 e a ligação deste à ficha DB9.

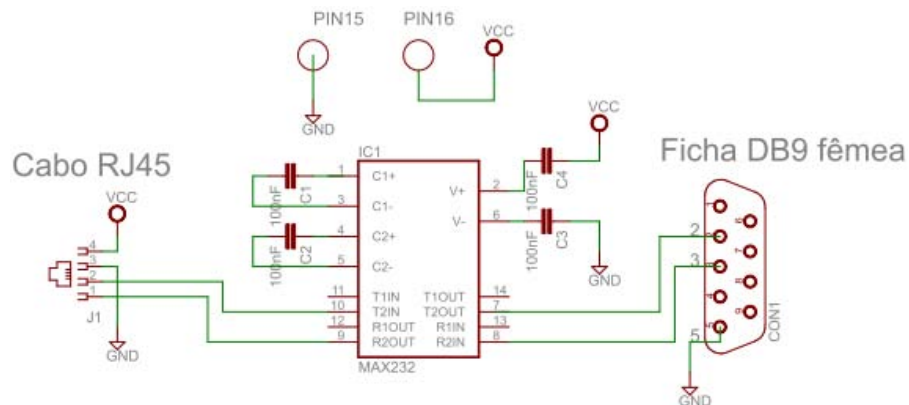


FIGURA 4-11 – ESQUEMA ELÉCTRICO DE LIGAÇÃO CABO RJ45 – MAX232 – FICHA FÊMEA DB9

As imagens 4.12 e 4.13 mostram a placa da MAX232 incorporada dentro da ficha fêmea DB9.

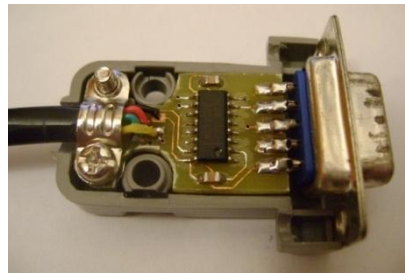


FIGURA 4-12 – TOP VIEW DA PLACA COM A MAX232 INCORPORADA NA FICHA FÊMEA DB9

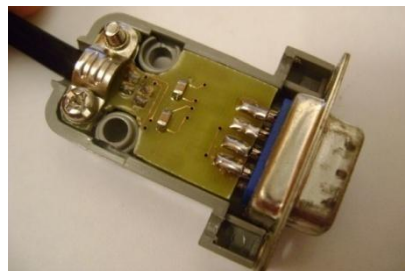


FIGURA 4-13 – BOTTOM VIEW DA PLACA COM A MAX232 INCORPORADA NA FICHA FÊMEA DB9

4.9. Placa PCB

A figura 4.14 apresenta a placa PCB do segundo protótipo para o Medidor.

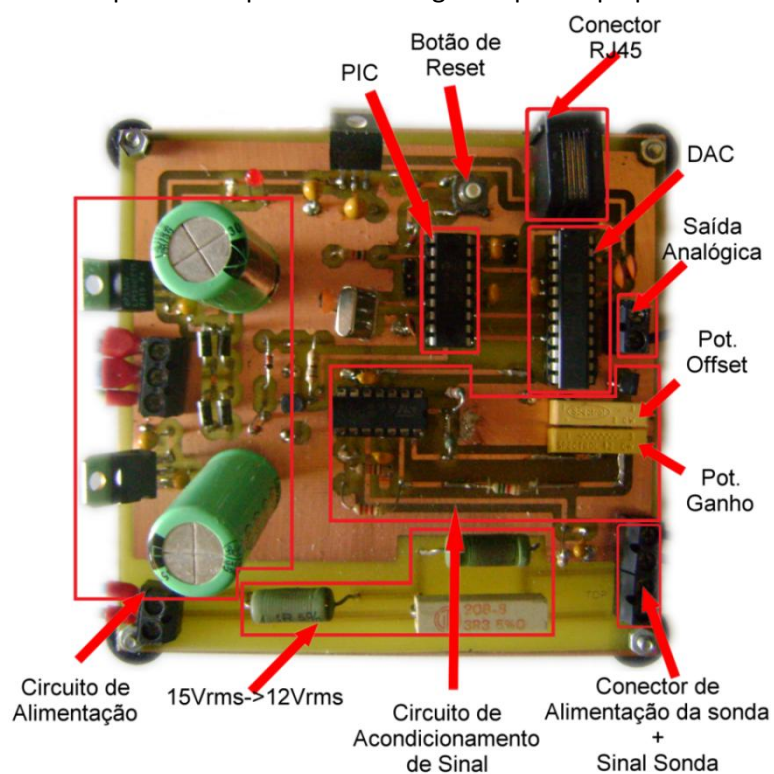


FIGURA 4-14 – PLACA PCB DO MEDIDOR

Capítulo 5

5. Software

5.1. Introdução

Neste capítulo é apresentado o fluxograma do código desenvolvido para ser implementado no microcontrolador. O código, quando implementado no microcontrolador, passará a ser o cérebro do Medidor de %O₂, pois este irá permitir ao microcontrolador interpretar o sinal proveniente da sonda lambda em termos de %O₂, converter esse valor no código utilizado pela DAC e, por fim, transmitir esse código por SPI para a DAC. Entretanto, é feita uma breve descrição das ferramentas utilizadas para programar o microcontrolador.

5.2. MPLAB® ICD 2 In-Circuit Debugger/Programmer

O MPLAB® ICD 2 In-Circuit Debugger/Programmer é um produto de baixo custo da Microchip, que em conjunto com a interface gráfica MPLAB IDE (*Integrated Development Environment*), permite programar e fazer o *debug* em tempo-real da maior parte dos PIC's e dsPIC's da tecnologia Microchip. É uma alternativa aos tradicionais ICE (*In-Circuit Emulators*).

O MPLAB ICD 2 liga-se ao PC através de uma interface USB ou RS-232 e pode ser ligado ao microcontrolador através de um conector ICD conforme indicado na figura 5.1 [22].



FIGURA 5-1 – ESQUEMA DE LIGAÇÃO PC – MPLAB ICD2 – PLACA COM MICROCONTROLADOR

O conector tem 5 pinos, dois dos quais estão conectados aos pinos de massa e de alimentação do microcontrolador, um terceiro está conectado ao pino /MCLR, estando os restantes ligados a um dos pares de pinos PGC/EMUC e PGD/EMUD existentes no microcontrolador. Estes dois últimos pinos são os responsáveis pelo *In Circuit Serial Programming™ (ICSP™)* e pelo *In-Circuit Debugger* dos dispositivos [22].

Deste modo, os programas podem ser facilmente carregados para o sistema *embedded*, executados em tempo real e serem examinados com detalhe pelas ferramentas de *debug* do MPLAB [22].

Vantagens [22]:

- Permite programar e fazer o *debug* de placas de uma linha de produção;
- Não requer sockets ou adaptadores muito caros;
- Consoante a aplicação, pode ser necessário mais algum hardware extra, mas o número de hardware necessário é muito reduzido.

Desvantagens [22]:

- Entre outras, o MPLAB ICD 2 usa alguns recursos do microcontrolador, tais como, pinos *I/O*, *program memory*, *data memory* e espaço da *stack*. Em consequência, algumas partes da aplicação desenvolvida para o sistema *embedded* não podem ser testadas e corrigidas.

Requisitos do PC que fará a comunicação com o MPLAB ICD 2 [22]:

- PC-compatible system com um processador da Intel Pentium®class, superior ou equivalente;
- 32MB de RAM disponíveis;
- 40MB de disco rígido disponíveis;
- Um porto RS-232 ou USB disponíveis;
- Microsoft®Windows®98, Windows NT®4.0, Windows 2000, Windows XP ou Windows Vista. O suporte USB pode ser limitado pelo sistema operativo Windows, particularmente o Windows 98/NT;
- CD-ROM drive para ser usado com o CD que o acompanha a embalagem do ICD2.

Por fim, convém referir que os microcontroladores Flash PICmicro que não suportam este produto são os PIC16F72/73/74/76/77/83/84ª [22].

As informações relativas à utilização da interface gráfica MPLAB IDE podem ser consultadas a partir do site

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en019469&part=SW007002

5.3. Fluxograma da aplicação desenvolvida

O fluxograma da figura 5.2, descreve, de modo mais simplificado, o código desenvolvido para implementar as operações necessárias à interpretação do sinal proveniente da sonda lambda em termos de %O₂, e transmitir esse valor por SPI para a DAC.

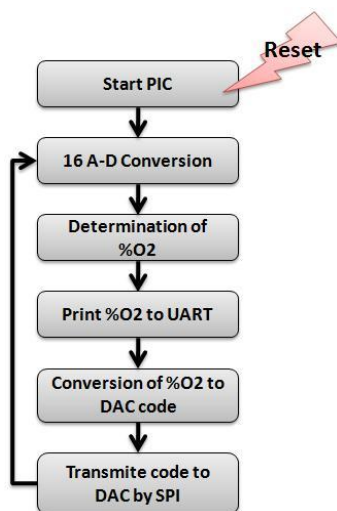


FIGURA 5-2 – FLUXOGRAMA DO CÓDIGO DESENVOLVIDO PARA O MEDIDOR

Como se observa no fluxograma (figura 5.2), após um reset, o microcontrolador passa para um bloco que se designa de “Start PIC”, onde são realizadas umas configurações iniciais, nomeadamente, a configuração dos registos associados aos módulos da ADC, SPI e *Timer*. Relativamente ao módulo da ADC, é configurado o formato dos dados apresentados pela ADC, a resolução usada, a frequência de amostragem, o *trigger* usado para iniciar a conversão, de entre outros detalhes. Relativamente ao módulo de SPI, é necessário configurar a frequência, a polaridade e a fase do sinal SCK, activar o modo *Master*, e definir o tamanho do *shift-register*, entre outros detalhes menos significativos. Relativamente ao *timer*, é configurado o período e o modo de 32bits. É ainda necessário configurar correctamente os pinos remapeados e definir os pinos que são de entrada e de saída, digitais e analógicos.

Após as configurações e as inicializações, o microcontrolador entra num ciclo sem fim, onde passará a executar as tarefas que lhe competem. Esse ciclo inicia-se com a recolha de 16 amostras. Com as 16 amostras é feita uma média, implementando, deste modo, um filtro digital que elimine algumas componentes de alta frequência do ruído.

Posteriormente, o valor resultante da média é usado para determinar a percentagem de oxigénio através da fórmula $O_{2_{combust\tilde{a}o}} = e^{\left(\frac{E-B}{M}\right)}$. Os valores de M e B já foram determinados na secção 4.6.2.3 e são respectivamente, -1021.9 e 5511.6. Note que, o valor apresentado vem multiplicado por 10 de modo a incluir a parte decimal, por exemplo, o valor 20,9% de O₂ apresenta-se como 209.

Em seguida, esse valor é colocado no Txbuffer da UART para ser transmitido para um PC. Se um PC estiver ligado ao microcontrolador, os valores de %O₂ podem ser visualizados no programa *Hyperterminal*. Aliás, o processo de calibração passa por ajustar os potenciômetros de ganho e de *offset* conforme os valores observados num *Hyperterminal* ou noutro programa equivalente.

Depois de transmitir o valor da %O₂ pela UART, esse valor terá que ser convertido para a codificação utilizada pela DAC. A AD420 coloca à sua saída uma corrente entre 4 e 20mA proporcional ao código digital (compreendido entre 0x0000 e 0xFFFF) que recebe do microcontrolador. Portanto, a conversão a realizar é uma simples regra de três simples. É de notar que há interesse em transmitir no sinal 4-20mA um valor acima dos 21% de O₂ de modo a ser possível detectar a descalibração do Medidor. Assim, o valor a enviar para o AD420 é calculado do seguinte modo:

$$DACval = \frac{\%O_2 \cdot (2^{16} - 1)}{220}$$

Por fim, o valor é colocado no buffer do SPI, o pino 12 é colocado no estado *low*, seguem-se 16 ciclos do sinal SCK, concluindo-se a transmissão com a colocação do pino 12 no estado *high*.

5.4. Frequência de amostragem

De acordo com o Teorema de Nyquist, a quantidade de amostras por unidade de tempo de um sinal, ou frequência de amostragem, deve ser maior que o dobro da frequência mais elevada, contida no sinal a ser amostrado, para que possa ser reproduzido integralmente sem erro de aliasing¹⁹.

A frequência máxima do sinal proveniente da sonda é muito baixa (praticamente só possui componente DC), como tal, não há grandes constrangimentos relativamente à frequência de amostragem. Arbitrou-se que a frequência de amostragem seria de 152Hz. O processo de amostragem é realizado com o auxílio do *timer 2/3* que faz, automaticamente por *hardware*, o *trigger* dos instantes de amostragem. Deste modo, garante-se que não há *jitter* de amostragem devido ao *software*.

¹⁹ Aliasing - É a distorção que ocorre num sinal amostrado quando a taxa de amostragem não respeita a taxa mínima conforme Nyquist, impedindo a correcta recuperação do sinal.

Capítulo 6

6. Resultados

6.1. Introdução

Nesta secção são apresentados os resultados obtidos com a sonda Lambda. Numa primeira fase, apresenta-se um gráfico que demonstra a resposta da sonda em função da $\%O_2$ à saída do primeiro andar de ganho. Numa segunda fase, são apresentados os valores de $\%O_2$ obtidos com o Medidor desenvolvido, comparando-os com os respectivos valores indicados pelo analisador de O_2 .

6.2. Alterações no esquema de montagem para a realização dos testes

Antes de passar à apresentação dos resultados, deve ser referido que, durante os ensaios, constatou-se existir alguma dificuldade em manter uma mistura gasosa com composição estável a passar pela sonda lambda e pelo analisador. Contribui, para dificultar as condições de recolha dos valores, o facto de o tempo de resposta e de estabilização do analisador serem muito mais longos que o tempo de resposta da sonda lambda. A combinação destes dois factores dificultou, durante uma primeira fase de ensaios, a recolha de alguma informação útil e rigorosa, pois os valores transmitidos pelo analisador e pelo multímetro oscilavam significativamente.

A fonte que gerava maior instabilidade na mistura foi identificada após algumas semanas de ensaios. Estava relacionada com um compressor, que injectava ar atmosférico para as instalações do DAO e que estava a ser utilizado neste trabalho para gerar misturas gasosas com diferentes teores de O_2 . O compressor tem um ciclo típico de esvaziamento e bombagem de ar, cuja duração depende da sua utilização para as diversas aplicações a decorrer no DAO, mas que, normalmente, é lenta. Quando a pressão no compressor chega a um limiar inferior, este entra num período de carregamento do seu depósito com ar. O período de carregamento é muito rápido.

Este comportamento era cíclico. Por sua vez, a resposta da sonda e do analisador também eram cíclicos e acompanhavam o mesmo ritmo do compressor, à excepção de um desfasamento. A relação que existe entre o comportamento do compressor e as respostas da sonda e do analisador, reside na pressão do ar injectado pelo compressor

que ia variando o caudal de ar atmosférico, e consequentemente, variava a quantidade de oxigénio da mistura.

Mediante estes factos, o esquema de montagem sofreu algumas alterações. Na figura 6.1 apresenta-se o esquema de montagem com as novas alterações.

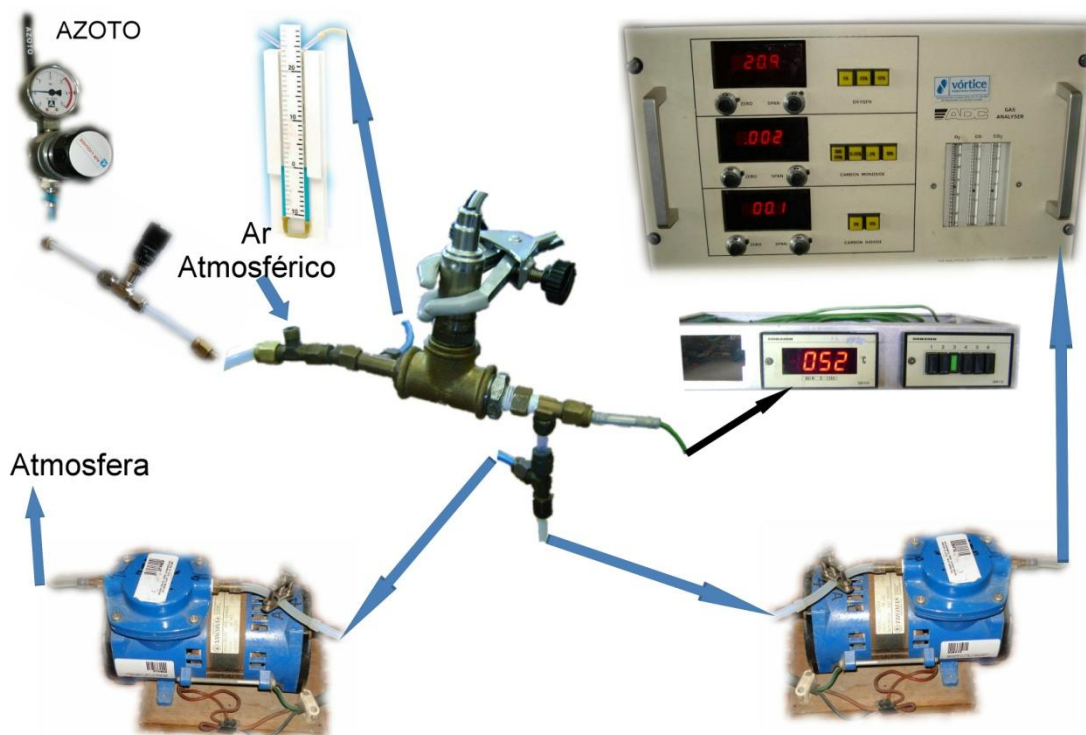


FIGURA 6-1 – ESQUEMA DE MONTAGEM COM AS NOVAS ALTERAÇÕES

6.2.1. Equipamento utilizado

- 1 Analisador de Oxigénio (1)
- 1 Termopar (2)
- 1 Tê Latão F/F com encaixe para a sonda (3)
- 2 Motores de ar (4)
- 1 Válvula de controlo de fluxo de ar (5)
- 1 Manómetro de coluna de ar com extremidade aberta (6)
- 1 Redutor de ar (azoto) (7)
- 1 Suporte para fixar a sonda

Os equipamentos representados na figura 3.3 foram gentilmente cedidos pelo Departamento de Ambiente e Ordenamento (DAO) da Universidade de Aveiro.

Relativamente ao esquema de montagem apresentado no capítulo 3, deixou-se de utilizar o ar atmosférico proveniente do compressor existente nas instalações do DAO. Para gerar várias misturas com diferentes teores de oxigénio, partiu-se da lei das nós – lei da continuidade:

“A soma de todos os caudais de saída tem de ser igual á soma de todos os caudais de entrada num sistema.”

Assim, utilizando uma abertura para deixar entrar ar atmosférico no sistema e mantendo um determinado caudal de saída através dos dois motores, consegue-se, por regulação da válvula que controla o caudal de azoto, controlar o caudal de ar atmosférico que entra pela abertura, pois a soma destes dois caudais tem de ser igual à soma dos caudais que passam pelos dois motores.

Ao sistema foi ainda acrescentado um termopar e um manómetro de coluna de água com uma extremidade aberta, para averiguar se as condições de temperatura e pressão no sistema se mantinham relativamente estáveis.

6.3. Apresentação de resultados

O gráfico 6.1 apresenta a resposta da sonda, para diferentes experiencias realizadas nos dias assinalados, em função da %O₂ após o primeiro op-amp do circuito da figura 4.5.

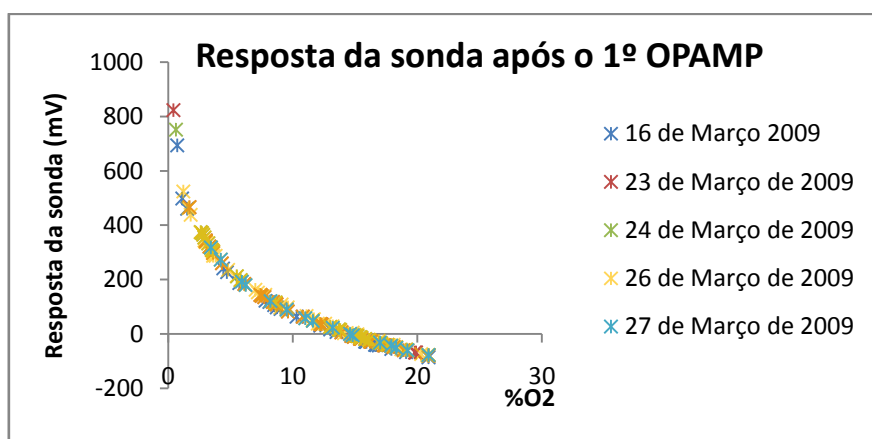


GRÁFICO 6.1 – RESPONDA DA SONTA À SAÍDA DO PRIMEIRO ANDAR DE GANHHO EM FUNÇÃO DA PERCENTAGEM DE OXIGÉNIO

Com base neste gráfico verifica-se que, nos ensaios realizados, a sonda apresentava uma boa repetibilidade²⁰.

²⁰ Repetibilidade – mede o grau de concordância entre os resultados de medições sucessivas da mesma grandeza, quando as condições de medição se mantêm constantes.

O gráfico 6.2 apresenta os valores indicados pelo Medidor de %O₂ desenvolvido neste trabalho após uma calibração da sonda e o respectivo valor indicado pelo analisador.

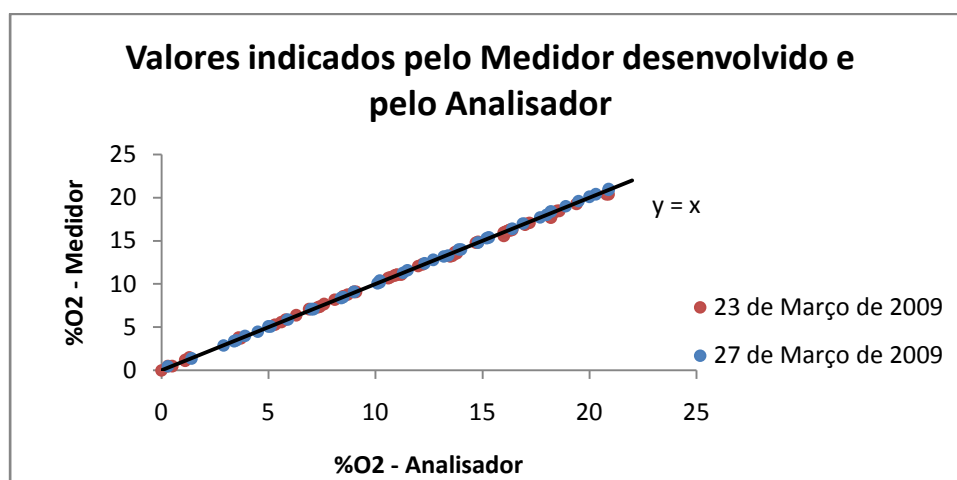


GRÁFICO 6.2 – VALORES OBTIDOS PELO MEDIDOR E PELO ANALISADOR

O gráfico 6.3 mostra a discrepância entre os resultados indicados pelo analisador e pelo Medidor em função da %O₂.

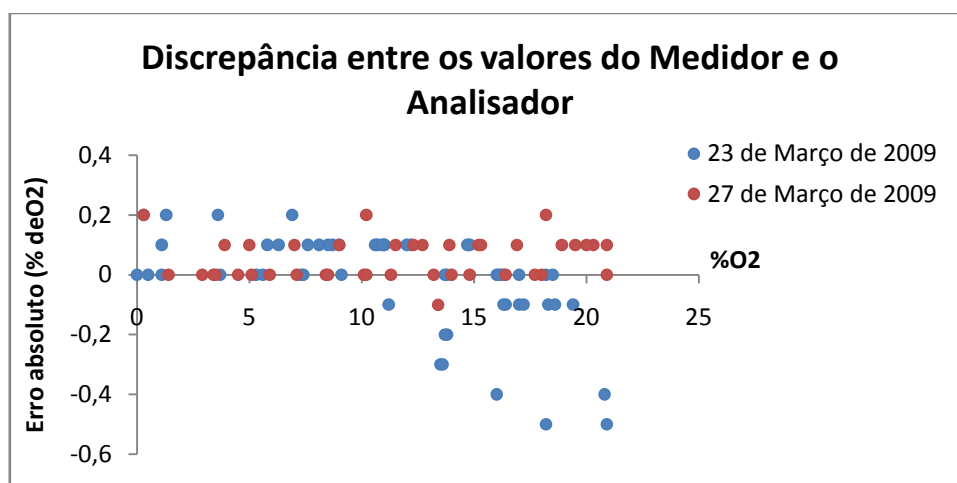


GRÁFICO 6.3 – DISCREPÂNCIA ENTRE OS VALORES INDICADOS PELO MEDIDOR FACE AOS VALORES INDICADOS PELO ANALISADOR

Como se pode observar nos gráficos 6.2 e 6.3, o Medidor desenvolvido apresenta bons resultados quando comparado com o valor indicado pelo analisador. Tal como seria de esperar, as maiores discrepâncias foram registadas para as misturas gasosas com teor de oxigénio elevado, devido à natureza logarítmica da sonda. O erro máximo em valor absoluto surge no ensaio realizado a 23 de Março de 2009 e tem amplitude de 5 décimas (0,5 %O₂). O erro médio da experiência realizada no dia 23 de Março é de 1 décima (0,1 %O₂) e o erro médio da experiência realizada no dia 27 de Março é de 0,6 décimas (0,06 %O₂).

Os resultados obtidos são satisfatórios e cumprem os requisitos e as especificações impostas, nomeadamente, resolução inferior a 0,2% de O_2 e erro absoluto não superior a 0,5% de O_2 . No entanto, o tempo despendido no processo de calibração é considerável, devido à sua natureza iterativa e à sensibilidade do utilizador, entre outros factores.

Mediante este facto, houve necessidade de encontrar uma solução que permitisse realizar a calibração da sonda em menos de 5 minutos.

Capítulo 7

7. Solução Final – Introdução da auto-calibração por software na aplicação desenvolvida

7.1. Introdução

Neste capítulo é apresentada a solução desenvolvida para diminuir o excessivamente longo tempo despendido na fase de calibração. A solução desenvolvida permite que o Medidor seja calibrado em menos de 5 minutos.

7.2. Calibração do Medidor de %O₂ por software

É sabido que, para determinar a equação de uma recta, é necessário conhecer dois pontos da recta.

É também sabido que a equação reduzida da recta é dada por $y = m \cdot x + b$, e que os parâmetros m e b são calculados como se segue:

$$m = \frac{y_2 - y_1}{x_2 - x_1}$$

$$b = y_1 - m \cdot x_1$$

Como já foi discutido no capítulo 4, a equação característica da sonda após o Bloco de Acondicionamento de Sinal, ou seja, à entrada da ADC, é dada por $V_{ADC} (\%O_2) = M \cdot \ln(\%O_2) + B$. Ora, esta é uma equação característica de uma recta. Conhecendo dois pontos, facilmente se determina o valor dos parâmetros M e B .

Após a determinação dos parâmetros M e B , os seus valores terão que ser guardados em memória flash (*Program Memory*) do microcontrolador, uma vez que uma falha de alimentação apaga o valor actual de todas as variáveis que se localizam na memória RAM, sendo estas inicializadas com o valor definido por defeito pelo programador. Assim, recorreu-se a instruções que permitam ler/escrever da/na memória *flash* em tempo de execução do microcontrolador.

Desta forma, consegue-se contornar o problema por Software, sendo apenas necessário adicionar um botão ao Hardware já desenvolvido, mais concretamente entre o pino 9 e a massa. A função desse botão é accionar a rotina de calibração.

7.2.1. Alterações ao Hardware

A figura 7.1 apresenta a placa PCB com o botão de calibração.

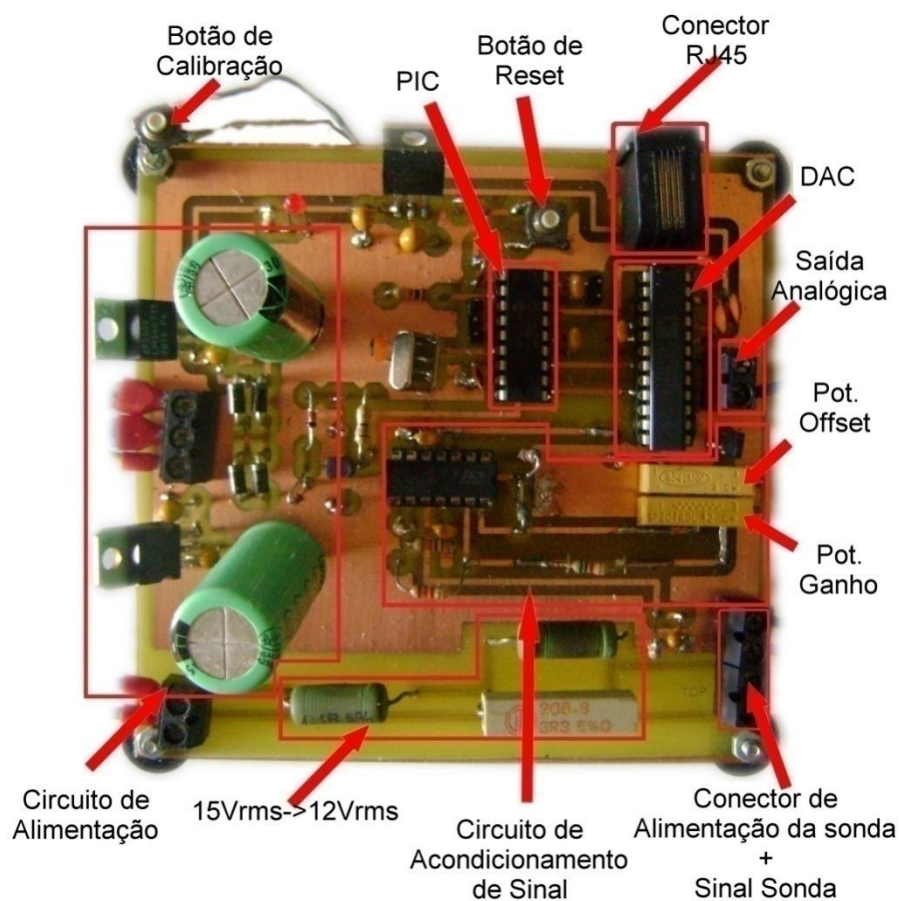


FIGURA 7-1 – PLACA PCB COM O BOTÃO DE CALIBRAÇÃO

7.2.2. Software

O código desenvolvido na primeira aplicação foi integralmente utilizado na segunda aplicação. Apenas foi necessário acrescentar ao código as instruções relacionadas com a calibração. O fluxograma da figura 7.2 ilustra o corpo da segunda aplicação desenvolvida, onde foi introduzida a auto-calibração.

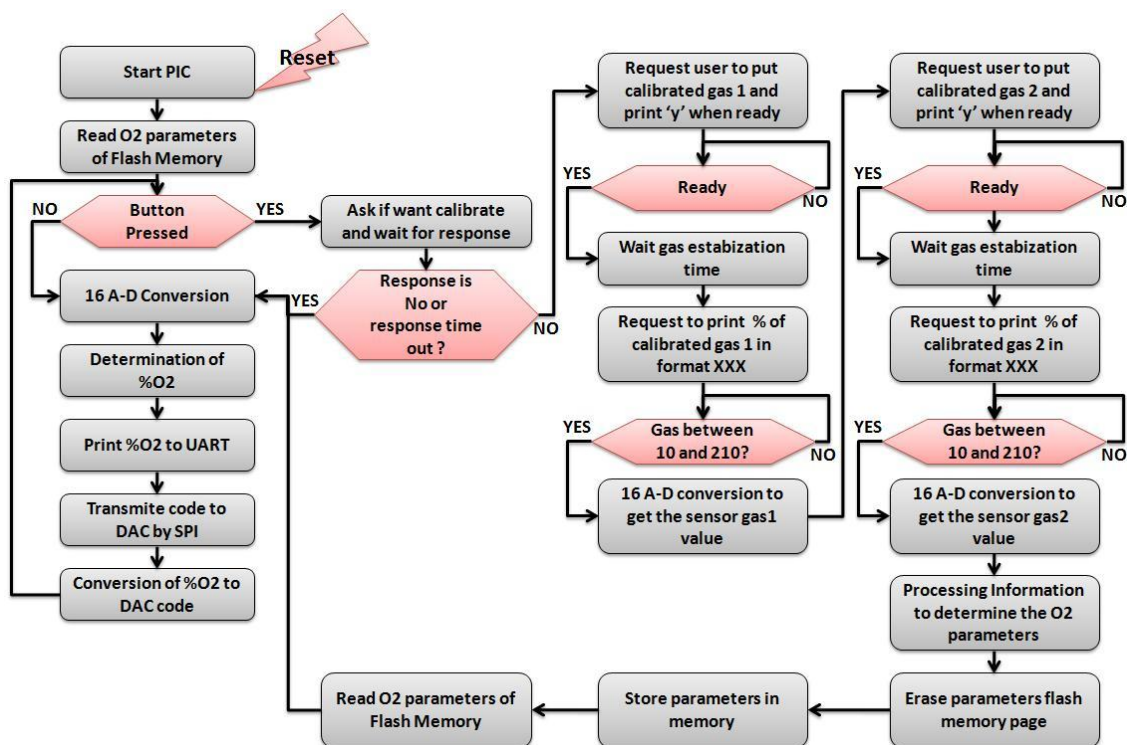


FIGURA 7-2 – FLUXOGRAMA DO CÓDIGO DO MEDIDOR COM CALIBRAÇÃO POR SOFTWARE

Como se pode observar no fluxograma, após as inicializações e as configurações e antes de entrar no ciclo infinito, foi necessário introduzir umas instruções que permitissem ler da memória *flash* para a memória RAM os parâmetros da recta. Na aplicação anterior estes parâmetros eram fixos e, como tal, as variáveis associadas eram automaticamente carregadas com o seu valor.

Verifica-se ainda que, logo no início do ciclo infinito e antes de recolher as 16 amostras, surgiu uma nova instrução que verifica se, entretanto, surgiu alguma interrupção accionada pelo botão de calibração.

Caso o botão não tenha sido pressionado, o microcontrolador passará a executar as tarefas normalmente, como executava na primeira aplicação. Caso tenha sido pressionado, o microcontrolador entra na “rotina” de calibração que a seguir se descreve.

1. Numa primeira fase o microcontrolador envia uma mensagem para o PC a questionar se deseja realmente calibrar a sonda, uma vez que o botão pode ter sido accionado despercebidamente por alguém ou pode ter sido accionado por lapso do utilizador. O microcontrolador entra num estado de bloqueio e permanece neste estado enquanto não receber um carácter específico (‘y’, ‘Y’, ‘n’ ou ‘N’) por parte do utilizador. No caso de o botão ter sido accionado despercebidamente, é muito provável que ninguém saiba que o Medidor de %O₂ tenha entrado num estado bloqueante e que esteja a aguardar a confirmação se deve ou não entrar na rotina de calibração. Foi necessário

introduzir um mecanismo de protecção contra estes eventos indesejados. Esse mecanismo foi implementado através do *timer* de 32 bits, que actua como temporizador. Estipulou-se que o utilizador teria cerca de 30 segundos para responder, e que após esse período, o microcontrolador interpretava a resposta como sendo “Não” e regressava à sua rotina habitual.

2. Caso o utilizador confirme o seu desejo em calibrar a sonda, o Medidor de %O₂ entra na rotina de calibração. Uma mensagem é enviada para o PC a indicar ao utilizador que coloque o primeiro gás padrão de calibração a passar pela sonda e para imprimir a tecla ‘y’ ou ‘Y’, quando o gás estiver a passar pela sonda. O microcontrolador volta a entrar num estado bloqueante, enquanto não receber a confirmação para continuar. Não foi implementado nenhum temporizador para este caso, uma vez que o utilizador mostrou interesse em calibrar a sonda na fase anterior. Caso pretenda desistir, poderá fazê-lo em qualquer instante pelo botão de *reset*.
3. Após a confirmação do utilizador a indicar que já colocou o primeiro gás de calibração, o microcontrolador envia uma mensagem para o PC a solicitar que aguarde. Um temporizador de cerca de dois minutos é activado, de modo a garantir que a mistura e a resposta da sonda estabilizem.
4. Depois de ter decorrido cerca de 2 minutos, após a confirmação do passo 2, será pedido ao utilizador para introduzir o valor de %O₂ do gás que está a passar pela sonda, pela seguinte ordem: dígito da dezena, o dígito da unidade e o dígito da parte decimal. O valor introduzido terá que se situar entre 210 e 010. O microcontrolador não passará ao passo seguinte, enquanto o valor introduzido não for válido.
5. Após a validação do valor introduzido, o microcontrolador recolhe 16 amostras e guarda o valor médio numa variável.
6. O Medidor repete os passos 2-5, substituindo o primeiro gás de calibração pelo segundo gás de calibração ou a atmosfera.
7. Depois de conhecidos os valores do primeiro e do segundo gás, assim como o valor médio do sinal à entrada da ADC para o primeiro e para o segundo gás, o microcontrolador irá determinar os valores dos parâmetros M e B de acordo com as fórmulas:

- $M = \frac{MediaGas\ 1 - MediaGas\ 2}{\ln(gas\ 1) - \ln(gas\ 2)}$
 - $B = MediaGas1 - M * \ln(gas1)$, onde *MediaGas1* e *MediaGas2* são respectivamente os valores médios do sinal à entrada da ADC quando a sonda está a monitorizar o primeiro e o segundo gás padrão de calibração; *gas1* e *gas2* são respectivamente o primeiro e o segundo gás padrão de calibração.
8. Após a determinação dos valores dos parâmetros, será feito um *erase* à página de memória onde estão guardados esses parâmetros. Como estes parâmetros são guardados na memória *flash*, ou seja, na memória do programa, teve-se o cuidado de colocar estes parâmetros numa página de memória isolada, de modo a não apagar, nem alterar pedaços de código.
 9. Os parâmetros são guardados na respectiva região de memória.
 10. Os parâmetros, depois de guardados na memória *flash*, são lidos para variáveis da memória de dados.
 11. O microcontrolador regressa à sua rotina habitual.

Todos os blocos do fluxograma foram testados individualmente, foram integrados e testados em conjunto.

7.3. Resultados obtidos

Nesta secção são apresentados os resultados obtidos com o Medidor de %O₂ com calibração por software.

Antes de apresentar os resultados, é de referir que a sonda foi calibrada por software, utilizando uma mistura de gás com 6% de O₂ e o ar atmosférico, como gases de referência para a calibração. Teve-se o cuidado de ajustar os potenciómetros com o mesmo valor de ganho e o mesmo valor de offset utilizado na experiência realizada a 27 de Março de 2009 com o intuito de verificar se os parâmetros M e B determinados pelo microcontrolador são próximos dos valores calculados manualmente na secção 4.6.2.3.

Utilizando o MPLAB ICD2 e uma das funcionalidades do interface gráfico MPLAB IDE, carregou-se para o PC os dados de memória do microcontrolador após a calibração, tendo-se verificado que os valores dos endereços de memória onde estão armazenados os parâmetros M e B, são 0xC61E9D19 (*float*) e 0x0000D566 (*long*), respectivamente (figura 7.3). Os valores destes parâmetros em decimal são

10151.2676 e 5463.0 e, portanto, verifica-se que estes valores estão próximos daqueles que foram calculados manualmente na secção 4.6.2.3.

LambdaInterface - MPLAB IDE v8.10 - [Program Memory]

File Edit View Project Debugger Programmer Tools Configure Window Help

Checksum: 0x14bb Release

| | Line | Address | Opcode | Label | Disassembly |
|--|------|---------|--------|-------|-------------|
| | 2042 | 0FF2 | FFFFFF | | nopr |
| | 2043 | 0FF4 | FFFFFF | | nopr |
| | 2044 | 0FF6 | FFFFFF | | nopr |
| | 2045 | 0FF8 | FFFFFF | | nopr |
| | 2046 | 0FFA | FFFFFF | | nopr |
| | 2047 | 0FFC | FFFFFF | | nopr |
| | 2048 | 0FFE | FFFFFF | | nopr |
| | 2049 | 1000 | 009D19 | | nop |
| | 2050 | 1002 | 00C61E | | nop |
| | 2051 | 1004 | 00D566 | | nop |
| | 2052 | 1006 | 000000 | | nop |
| | 2053 | 1008 | 0088AF | | nop |
| | 2054 | 100A | 00CCCC | | nop |
| | 2055 | 100C | 00C646 | | nop |
| | 2056 | 100E | 00B4CF | | nop |
| | 2057 | 1010 | 00E029 | | nop |
| | 2058 | 1012 | 00043E | | nop |
| | 2059 | 1014 | 004E94 | | nop |
| | 2060 | 1016 | 002181 | | nop |
| | 2061 | 1018 | 003737 | | nop |
| | 2062 | 101A | 008637 | | nop |
| | 2063 | 101C | 00D523 | | nop |
| | 2064 | 101E | 00366F | | nop |

FIGURA 7-3 – CONSULTA DOS DADOS DE MEMÓRIA DO MICROCONTROLADOR NOS ENDEREÇOS ONDE RESIDEM OS PARÂMETROS M E B

No gráfico 7.1 estão apresentados os valores de tensão, medidos à saída do primeiro andar de ganho do circuito de acondicionamento de sinal (figura 4.5) em função da percentagem de oxigénio.

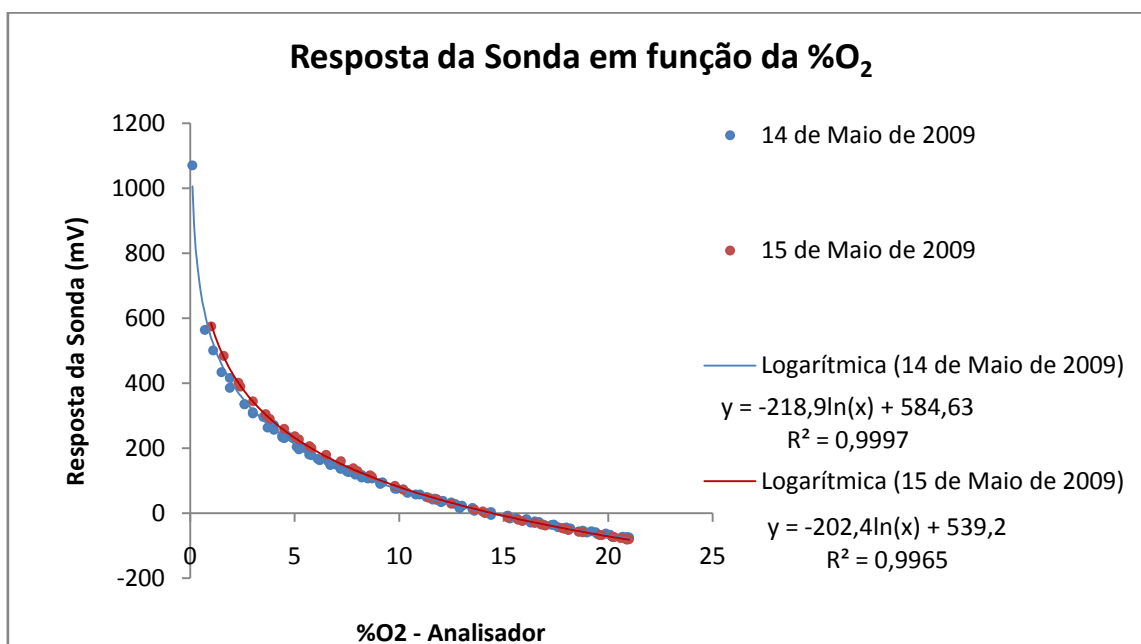


GRÁFICO 7.1 – RESPOSTA DA Sonda EM FUNÇÃO DA PERCENTAGEM DE OXIGÉNIO

Com base neste gráfico verifica-se que, nos ensaios realizados, a sonda apresentava uma ligeira discrepância entre os ensaios realizados a 14 de Maio e os ensaios realizados a 15 de Maio. Esta discrepância justifica-se pelas ligeiras variações da pressão atmosférica entre estes dois dias, uma vez que tanto o analisador como a sonda lambda sofrem pequenas influências das variações da pressão atmosférica.

No gráfico 7.2 estão apresentados os valores obtidos com o Medidor desenvolvido, calibrado por software utilizando como referencia o ar atmosférico e um gás com 4,4% de O₂, e os respectivos valores indicados pelo analisador.

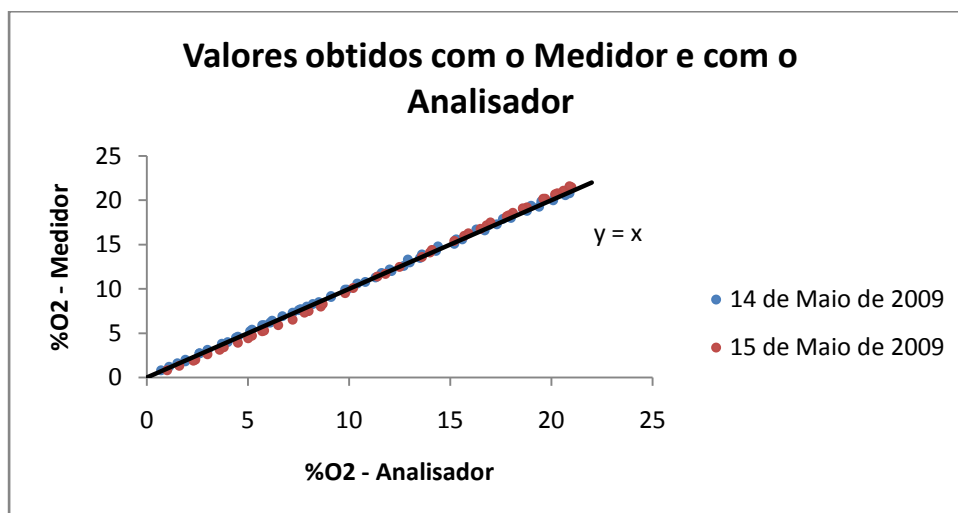


GRÁFICO 7.2 – VALORES OBTIDOS PELO MEDIDOR E PELO ANALISADOR

O gráfico 7.3 mostra a discrepância entre os resultados indicados pelo analisador e pelo Medidor em função da %O₂.

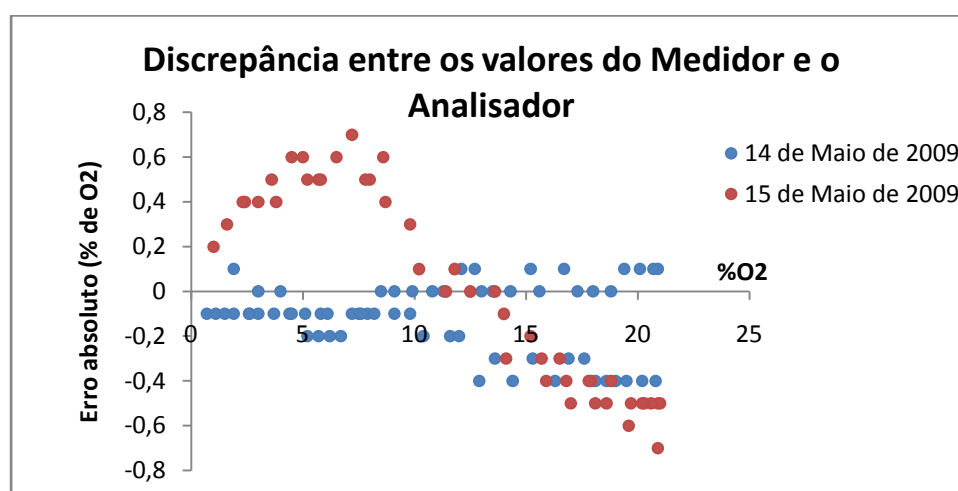


GRÁFICO 7.3 – DISCREPÂNCIA ENTRE OS VALORES INDICADOS PELO MEDIDOR FACE AOS VALORES INDICADOS PELO ANALISADOR

Como se pode observar nos gráficos 7.2 e 7.3, o Medidor desenvolvido apresenta resultados bastante satisfatórios, contudo o erro máximo é superior a 0,5% de O₂. As maiores discrepâncias entre os valores indicados pelo analisador e pelo Medidor desenvolvido registam-se no dia 15 de Maio por se ter utilizado nesses ensaios a mesma calibração do dia anterior e, por isso, essas discrepâncias reflectem a variação da pressão atmosférica relativamente ao dia anterior.

Capítulo 8

8. Conclusões e Trabalho futuro

O principal objectivo deste projecto visava desenvolver um aparelho de medida simples, robusto, de fácil calibração, que indicasse a percentagem de oxigénio contida numa mistura gasosa, para ser essencialmente aplicado no controlo de qualquer processo de combustão industrial.

O Medidor de %O₂ foi recebendo diversos ajustamentos ao longo do processo de desenvolvimento, devido aos problemas que foram identificados durante os ensaios realizados e de modo a torná-lo mais simples, mais preciso, e de fácil calibração.

O primeiro protótipo desenvolvido não tomava em consideração as variações da resposta da sonda face às pequenas mudanças das condições ambientais, visto que, foi projectado especificamente para os resultados recolhidos num ensaio concreto, realizado sobre as condições ambientais desse dia e não era previsto que a resposta da sonda pudesse variar com ligeiras mudanças das condições ambientais.

Desenvolveu-se um segundo protótipo que acomodava facilmente a não reprodutibilidade da sonda face às variações das condições ambientais, através do ajuste de dois potenciómetros e com auxílio de um PC. Os resultados obtidos com este protótipo eram bastante satisfatórios, porém, o tempo despendido no processo de calibração é considerável, devido à sua natureza iterativa e à sensibilidade do utilizador.

Por fim, desenvolveu-se um terceiro protótipo que permite facilmente calibrar o Medidor de %O₂ em menos de 5 minutos com auxílio de um PC. Os resultados obtidos mostraram-se também satisfatórios.

Reflectindo sobre o trabalho realizado, as maiores dificuldades encontradas prenderam-se não com a sua concepção, mas sim com a construção de um esquema de montagem mecânica, que permitisse gerar diferentes misturas com diferentes teores de oxigénio, e que garantisse que estas misturas fossem estáveis, durante o processo de recolha de dados, assim como garantir que a temperatura e a pressão parcial dos gases se mantivessem constante ao longo do ensaio, de modo a obter resultados fidedignos.

Do estudo realizado no âmbito deste projecto, é possível concluir que um produto como, este será certamente uma mais-valia para os fabricantes de caldeiras industriais, para os proprietários e também para o meio ambiente.

Com este produto, os fabricantes das caldeiras industriais poderão equipar as suas caldeiras, com um sistema económico que permite obter um maior rendimento das mesmas e cumprir as normas ambientais sobre a emissão de poluentes. Deste modo, espera-se alargar o controlo da emissão de poluentes até às mais pequenas indústrias de combustão, o que até ao momento era praticamente impossível, devido aos elevados custos dos produtos similares existentes no mercado. Assim, também os proprietários passarão a encontrar no mercado caldeiras, que cumprem as normas ambientais, por um custo mais acessível.

Como trabalho futuro, aguarda-se por uma oportunidade de poder implementar o Medidor de %O₂ em situação real, para avaliar o desempenho da sonda em condições reais, onde a pressão e a temperatura não são tão estáveis como as de laboratório.

Além disso, é também necessário desenvolver um esquema de montagem que permita variar a temperatura dos gases entre 0°C e 600°C, mantendo a pressão constante, de modo a avaliar a influência da temperatura na sonda.

Bibliografia:

- [1] PINHEIRO, Paulo César da Costa e VALLE, Ramon Molina. Controle de Combustão: “Otimização do Excesso de ar”. In CONGRESSO DE EQUIPAMENTOS E AUTOMAÇÃO DA INDÚSTRIA QUÍMICA E PETROQUÍMICA, 09-11 Agosto 1995, Rio de Janeiro, RJ, Anais...;
- [2] LIMA, Lutero Carmo et al. “Construção e avaliação de um monitor de combustão industrial”. In Rev. Tecnol. Fortaleza, v. 28, n. 1, p. 77-84, jun. 2007, <http://www.unifor.br/notitia/file/1306.pdf>;
- [3] ZAJAC, Magdalena Gracyk e INOESCU, Emanuel. “Practical Materials Science II”, Universidade Técnica de Darmstadt, http://www1.tu-darmstadt.de/fb/ms/studium/lab_course_LambdaProbe.pdf, consultado pela última vez a 5 de Maio de 2009;
- [4] Bosch-Motorsport – <http://www.bosch-motorsport.de/content/language2/html/2886.htm>, consultado pela última vez a 5 de Maio de 2009;
- [5] Technical Data, “Lambda” Oxygen Sensors, Type LSM 11, http://www.bosch.se/content/language1/downloads/Sensorkatalog_sidor_58_till_60.pdf, consultado pela última vez a 12 de Maio de 2009;
- [6] Wikipédia: “Sonda Lambda” – http://pt.wikipedia.org/wiki/Sonda_lambda, consultado pela última vez a 5 de Maio 2009;
- [7] Technical Data, Lambda Probe LS1, <http://www.lamtec.de/v2/Datenblatt/TD-LS1-DLT6005-97-aE-0001.pdf>, consultado pela última vez a 5 de Maio de 2009;
- [8] Technical Data, Lambda Transmitter LT1, <http://www.lamtec.de/v2/Datenblatt/TD-LT1-DLT-6062-06-aE-0040.pdf>, consultado pela última vez a 5 de Maio de 2009;
- [9] Technical Data, Lambda Transmitter LT2, Lambda Probe LS2, <http://www.lamtec.de/v2/Datenblatt/TD-LT2-DLT6063-07-bE-0125.pdf>, consultado pela última vez a 5 de Maio de 2009;
- [10] Technical Data, Lambda Transmitter LT10 for O₂ measurement and process monitoring, <http://www.lamtec.de/v2/Datenblatt/TD-LT10-DLT6210-06-bE-0010.pdf>, consultado pela última vez a 5 de Maio de 2009;
- [11] ECM EGR 5230, “Exhaust Gas Recirculation(EGR), Lambda, AFR, and O₂ Analyser”, http://www.csm.de/download/pdf/ECM_EGR5230.pdf, consultado pela última vez a 5 de Maio de 2009;
- [12] Macolis – <http://www.macolis.pt/index.php?chave=rendimento&op=pesquisa>, consultado pela última vez a 5 de Maio de 2009;
- [13] Sensor de Oxigénio de Alta temperatura – Teoria de Operação, <http://www.metaltech.com.br/teoriase.html>, consultado pela última vez a 24 de Junho de 2009;

- [14] Oxygen Sensors, Service Tech Magazine/May 2001,
<http://turbobricks.com/resources/O2sensors.pdf>, consultado pela última vez a 5 de Maio de 2009;
- [15] Section A - Product Information,
http://www.bosch.com.au/content/language1/downloads/Section_A.pdf, consultado pela ultima vez a 24 de Junho de 2009;
- [16] Datasheet Rev F, 10/1999, "Serial Input 16-Bit 4mA – 20mA, 0mA – 20mA DAC AD420",
http://www.analog.com/static/imported-files/data_sheets/AD420.pdf, consultado a 5 pela última vez a 5 de Maio de 2009
- [17] Farnell Components S.L., <http://pt.farnell.com/>, Maio 2009;
- [18] PIC24H Family Datasheet,
<http://www.alliedelec.com/Images/Products/Datasheets/BM/microchip/383-0636.pdf>, consultado pela ultima vez a 24 de Junho de 2009;
- [19] TL431 Programmable Shunt Regulator, Fairchild Semiconductor Corporation (2003),
<http://www.fairchildsemi.com/ds/TL/TL431A.pdf>, consultado pela ultima vez a 24 de Junho de 2009;
- [20] TL081, TL084, TL084A, TL084B, General PurposeJ-FET QUAD OPERATIONAL AMPLIFIERS, STMicroelectronics (1999)
<http://www.datasheetcatalog.org/datasheet/texasinstruments/tl084.pdf>, consultado pela ultima vez a 5 de Maio de 2009;
- [21] Wikipédia : "SPI", http://en.wikipedia.org/wiki/Serial_Peripheral_Interface_Bus, consultado pela última vez a 6 de Maio de 2009
- [22] MPLAB® ICD 2 In-Circuit Debugger/Programmer – ref DS51264B,
<http://ww1.microchip.com/downloads/en/devicedoc/51264B.pdf>, consultado pela última vez a 6 de Maio de 2009

ANEXO A – Programa Principal

```

/*****<*****
*FileName: mainO2v5.c
*Author Almerindo Oliveira Paiva      Date 20/02/09
*~~~~~
*Last modification at 11/06/2009
*~~~~~
*This code implements an application developed at University of Aveiro
*to calculate the %O2 of a gas mixture based on a LSM11 Lambda Sensor
*from Bosch Technology.
*This application will convert a signal which varies logarithmically
*with the %O2 into a current signal which should vary linearment
*between 4mA and 20mA.
*This application provides the user with a calibration of the probe
*by Software
*****/

#include "p24Hxxx.h"
#include "uart.h"
#include "device_driversPIC24Hv5.h"
#include "spi.h"
#include "adc.h"
#include <stdio.h>
#include "timer.h"
#include "typeDef.h"
#include "rtspApi.h"
#include "testData.h"
_FWDT(FWDTEN_OFF);
#define FCY 20000000
#define BAUDRATE 38400
#define BRGVAL ((FCY/BAUDRATE)/16)-1
#define SS_TIME 16*16

/***** Consts *****/
const char Calstr [] = "You want calibrate?";
const char PressYNstr [] = " Press Y/N\n\r";
const char PressYstr [] = "press y\n\r";
const char newLinestr [] = "\n";
const char PutGasstr []="PUT gas ";
const char ANDstr []=" and ";
const char PrintO2str []= "Print percentage of O2 of gas ";
const char valorstr []= "valor -> ";
const char Nivelstr []= "Nivel -> ";
const char Formatstr []= "Format XXX\n\r";
const char PleasWaitstr []="Please Wait\n\r";

/***** Oscillator Configurations *****/
// External Oscillator
_FOSCSEL(FNOSC_PRIPLL);          // Primary (XT, HS, EC) Oscillator with PLL
_FOSC(FCKSM_CSDCMD & OSCIOFNC_OFF & POSCMD_XT);
// Clock Switching and Fail Safe Clock Monitor is disabled
// OSC2 Pin Function: OSC2 is Clock Output
// Primary Oscillator Mode: XT Crystanl

/***** Global variables *****/
unsigned int Timeout;
unsigned int NewRxChar;
unsigned int CALIBRA=0;
unsigned int rowMirrorBuff[64];
// RTSP variable
unsigned int nvmAdr,nvmAdu,nvmAdrPageAligned,nvmRow,nvmSize;

/***** Declare external interrupt ISRs *****/
void __attribute__((__interrupt__)) _T3Interrupt(void);
void __attribute__((__interrupt__)) _CNInterrupt(void);
void __attribute__((__interrupt__,__no_auto_psv)) _UIRXInterrupt(void);

/***** other functions *****/
unsigned int GetGasX(unsigned int xxx);
void ClearRxBuff(void);

int main(void)

```

```

{
    unsigned int i, temp, dados, media, CALIBRATION, DataOut, Gas1;
    unsigned int Gas2, Gas1N, Gas2N;
    unsigned long data;
    char cal;
    float Lgas1, Lgas2, difGL, b;
    float mx;
    float *mptr = (float *) &rowMirrorBuff;
    int difGN;
    long bb;
    long *bptr = (long *) &rowMirrorBuff+1;
    ConstInMem parameters;

    // Configure Oscillator to operate the device at 40Mhz
    // Fosc= Fin*M/(N1*N2), Fcy=Fosc/2
    // Fosc= 8M*40(2*2)=80Mhz for 8M input clock
    PLLFBD=38;          // M=40
    CLKDIVbits.PLLPOST=0;  // N1=2
    CLKDIVbits.PLLPRE=0;  // N2=2
    RCONbits.SWDTEN=0;    /* Disable Watch Dog Timer*/
    PSVPAG = 0;
    CORCONbits.PSV=1;
    while(OSCCONbits.LOCK!=1) { }; /* Wait for PLL to lock*/

    U1BRG = BRGVAL ;      /* BAUD Rate Setting of Uart2 */
    StartPIC();           /*Initializations & Configurations*/
    CALIBRATION=0;
    nvmmAdu=__builtin_tblpage(&MyRowDataInFlash[0]);
    nvmmAdr=__builtin_tbloffset(&MyRowDataInFlash[0]);
    nvmmAdrPageAligned=nvmmAdr & 0xFC00;  // Get the Flash Page Aligned address
    nvmmRow=((nvmmAdr>>7) & 7);          // Row in the page
    nvmmSize=64;

    // Clear Row Mirror Buffer
    for (i=0; i<64;i++) {
        rowMirrorBuff[i]=0;
    }

    //Read parameters M & B of Program memory
    temp = flashPageRead(nvmmAdu,nvmmAdrPageAligned, &rowMirrorBuff);
    mx= *mptr;  //Load parameter M to mx
    bb= *bptr;  //Load parameter B to bb

    while(1)
    {
        //Calibration Button Pressed???
        if(CALIBRA==1)    //yes it was!
        {
            CALIBRATION=CALIBRATION+1;  //this variable is used only to
            //protect for multiple CN interrupts generated by pressing the button
        }
        if(CALIBRATION==2)  //only one interrupt "rotine" will be executed
        {
            AD1CON1bits.ADON=0;  //close ADC module
            PutStr((char *)Calstr); //ask user if he want calibrated
            Timeout=0;
            ConfigTMR(0x001f0d18); //configure timeout to approximately 30s
            cal='.';

            //PIC will be blocked until user press y or n or until timeout is reached
            while (cal != 'y' && cal != 'Y' && cal != 'n' && cal != 'N')
            {
                PutStr((char *) PressYNstr);
                ClearRxBuff();          //clear Rx Buff
                NewRxChar=0;
                while(NewRxChar == 0 && Timeout==0)
                {
                }
                //if timeout is reached it will considered that response is no
                if (Timeout==1)
                {
                    cal = 'n';
                }
                else cal=ReadUART1();
            }
        }
    }
}

```

```

CloseTimer23();
//User want make a calibration
if (cal == 'y' || cal=='Y')
{
    /* GET (%O2, SensorVoltage) Point of Gas1 */
    Gas1=GetGasX(1); //Get %O2 of first calibrated gas
    Gas1N=AdcMean16(); // Get SensorVoltage for that gas
    PutStr((char *) Nivelstr);
    PutInteger(Gas1N);
    PutChar('\n');
    PutChar('\r');
    AD1CON1bits.ADON=0;

    /* GET (%O2, SensorVoltage) Point of Gas2 */
    Gas2=GetGasX(2); //Get %O2 of second calibrated gas
    Gas2N=AdcMean16(); // Get SensorVoltage for that gas
    PutStr((char *) Nivelstr);
    PutInteger(Gas2N);
    PutChar('\n');
    PutChar('\r');
    AD1CON1bits.ADON=0;

    //Now it will calculated the M & B parameters
    //the equation os one straight line is y=m*x+b
    //m=(SensorVoltage1-SensorVoltage2)/(ln(% O2gas1)-ln(% O2gas2))
    //b=SensorVoltage1-m*ln(% O2gas1)
    Lgas1=log(Gas1);
    Lgas2=log(Gas2);
    difGL=Lgas1-Lgas2;
    difGN=Gas1N-Gas2N;
    mx=((float)difGN)/difGL;
    b=mx*Lgas1;
    b=((float)Gas1N)-b;
    bb=(long)b*10;
    mx=mx*10;
    //write parameters in memory
    parameters.m=mx;
    parameters.b=bb;
    temp = flashPageErase(nvmAdru,nvmAdrPageAligned);
    temp = flashPageWrite(nvmAdru,nvmAdrPageAligned,&parameters);
    // Clear Row Mirror Buffer
    for (i=0; i<64;i++) {
        rowMirrorBuff[i]=0;
    }
    //Read parameters from Program Memory
    temp = flashPageRead(nvmAdru,nvmAdrPageAligned, &rowMirrorBuff);
    mx= *mptr;
    bb= *bptr;
}
CALIBRATION=0;
CALIBRA=0;
IEC1bits.CNIE = 1;
}

media=AdcMean16(); //Get the mean of 16 A-D conversion
dados=processamento(media,mx,bb); //process information to obtain %O2
PutInteger(dados);
PutChar('\n');
PutChar('\r');

//conversion of %O2 value to DAC code
data=((unsigned long)((unsigned long)dados*(unsigned long)65535);
data=data/220;
DataOut=(unsigned int) data;
//transmit by SPI the %O2 to DAC
SPI1STATbits.SPIEN=1;
SPI1BUF=DataOut;
while(SPI1STATbits.SPITBF==1);
LATBbits.LATB9=0;
for(i=0;i<33;i++)
{}
LATBbits.LATB9=1;
}

```

```

}

//Clear RX buffer to eliminated the preview received chars
void ClearRxBuff(void)
{
    unsigned char rubbish;

    if(U1STAbits.OERR == 1)
    {
        U1STAbits.OERR = 0;
    }

    while(U1STAbits.URXDA==1)
    {
        rubbish=ReadUART1();
    }
}

//Request the user to print th %O2 of the calibrated gas and returns the value
unsigned int GetGasX(unsigned int xxx)
{
    unsigned int gasx;
    unsigned int k, UN, TE, HU;
    char Buf[3];
    char PressY;

    //Request user to print y when the calibrated gas is ready in the system
    PutStr((char *) PutGasstr);
    PutInteger(xxx);
    PutStr((char *) ANDstr);
    PutStr((char *) PressYstr);

    ClearRxBuff();
    //wait for confirmation
    while(1)
    {
        NewRxChar = 0;
        while(NewRxChar == 0)
        {
        }
        PressY= ReadUART1();
        if(PressY == 'y' || PressY == 'Y') break;
        else PutStr((char *) PressYstr);
    }

    PutStr((char *) PleasWaitstr);
    Timeout=0;
    ConfigTMR(0x007f0d18); //wait approximately 2min for gas estabilization
    while(Timeout==0)
    {
    }
    CloseTimer23();

    //Request the user to print th %O2 of the calibrated gas
    PutStr((char *) PrintO2str);
    PutInteger(xxx);
    PutChar(' ');
    while (1)
    {
        PutStr((char *) Formatstr);
        ClearRxBuff();

        for(k=0;k<3;k++)
        {
            NewRxChar = 0;
            while(NewRxChar==0)
            {
            }
            Buf[k]= ReadUART1();
        }
        UN=Buf[2]-'0';
        TE=Buf[1]-'0';
        TE=TE*10;
        HU=Buf[0]-'0';
        HU=HU*100;
        gasx=HU+TE+UN;
    }
}

```

```

        PutInteger(gasx);
        PutChar('\n');
        PutChar('\r');
        if(gasx>9 && gasx<211) break; //validate the information
    }

    return gasx;
}

void __attribute__((interrupt, no_auto_psv)) _CNInterrupt(void)
{
    IFS1bits.CNIF = 0; //Clear the INT0 interrupt flag or else
                        //the CPU will keep vectoring back to the ISR
    CALIBRA=1;
}

void __attribute__((__interrupt__,no_auto_psv)) _T3Interrupt(void)
{
    IFS0bits.T3IF = 0; //Clear Timer3 interrupt flag
    Timeout=1;
}

void __attribute__((__interrupt__,no_auto_psv)) _U1RXInterrupt(void)
{
    IFS0bits.U1RXIF = 0; // clear TX interrupt flag
    NewRxChar=1;
}

```

ANEXO B – Device-Drivers do Programa Principal

/******

* FileName: device_driversPIC24Hv5.c

* Author Almerindo Oliveira Paiva Date 20/02/09

*Last modification at 11/06/2009

* This file have the device-drivers that supports the mainO2v5.c file.

*****/

#include "device_driversPIC24Hv5.h"

#include <math.h>

#include "adc.h"

#include "timer.h"

#include "uart.h"

#include <stdio.h>

void ConfigUART()

```
{
    //configure pin 14 as RX
    RPINR18bits.U1RXR = 14; // RX
    //configure pin 15 as TX
    RPOR7bits.RP15R = 3;
    U1MODE= 0x8800;      // 8-bit data, no parity, one stop bit
    IEC0bits.U1RXIE=1;   //enable RX interruption
    U1STAbits.UTXEN = 1; // Enable UART Tx
    IPC2bits.U1RXIP = 0x02; // Set Rx Interrupt Priority Level
    IFS0bits.U1RXIF = 0;  // Clear RX Interrupt Flag
    AD1PCFGLbits.PCFG7=1;
}
```

void PutInteger (unsigned int valInt)

```
{
    unsigned int aux;
    unsigned int righthZero=0;

    aux = valInt/10000;
    if (aux==0) righthZero = 1;
    else PutChar('0' + aux);

    valInt=valInt% 10000;
    aux = valInt/1000;

    if (righthZero==0 || aux!=0)
    {
        PutChar('0' + aux);
        righthZero=0;
    }

    valInt=valInt% 1000;
    aux = valInt/100;

    if (righthZero==0 || aux!=0)
    {
        PutChar('0' + aux);
        righthZero=0;
    }

    valInt = valInt% 100;
    aux = valInt/10;

    if (righthZero==0 || aux!=0)
    {
        PutChar('0' + aux);
        righthZero=0;
    }

    aux = valInt% 10;

    PutChar('0' + aux);
}
```

void PutStr(char * st)

```

{
    char *s=st;

    while(*s!='\0')
    {
        PutChar(*s);
        s++;
    }
}

void PutChar(char Char)
{
    //putchar(Char);
    while (U1STAbits.UTXBF); //transmit buffer is full
    U1TXREG = Char;
}

void ConfigADC()
{
    TRISAbits.TRISA1=1;
    TRISAbits.TRISA0=1;
    //12bit, GP timer 3 compare ends and starts conversion, sampling starts
    //immediately after last conversion, ADC off
    AD1CON1= 0x8446;

    AD1CON2= 0x2000; //ADref+> external Vref+
    AD1CON3=0x0002; //ADC conversion clock derives from system clock

    ADPCFG=ENABLE_AN1_ANA; //Enable AN1 as analog input
    AD1CSSL=0; //NO select any channel of ADC to scan
    AD1CHS0bits.CH0NA=0;
    AD1CHS0bits.CH0SA=1;
    IFS0bits.AD1IF = 0; // Clear AD1 Interrupt Flag
}

void ConfigTMR(unsigned long period)
{
    TMR2 = 0; // Reset Timer2 to 0x0000 */
    TMR3 = 0; // Reset Timer3 to 0x0000 */
    PR2 = period; // assigning Period to PR2 */
    PR3 = period>>16; // Period to PR3 Register */
    //enable timer, prescaler 1:256, 32 bit mode, timer 2 clock source
    //is internal clock
    T2CON=0x8038;
    IPC2bits.T3IP = 0x02; // Set Timer3 Interrupt Priority Level
    IFS0bits.T3IF = 0; // Clear Timer3 Interrupt Flag
    IEC0bits.T3IE = 1; // Enable Timer3 interrupt
}

void ConfigSPI()
{
    //configure pin 8 as SCK
    RPOR4bits.RP8R=0b01000;
    //configure pin 7 as SDO
    RPOR3bits.RP7R=0b00111;

    //Configure polarity as
    SPI1STATbits.SPISIDL=0;
    SPI1STATbits.SPIROV=0;

    //communication is word-wide, master mode, idle state for clock is low level
    //enable internal SPI clock,SDO is controled by the module,Serial output data
    //changes on transition from active clock,primary prescaler 16:1, secondary
    //prescaler is 1:1.
    SPI1CON1=0x053d;
    SPI1CON2=0x0000;
}

void StartPIC()

```

```

{
    TRISB= (TRISB & 0x7c7f) | 0x4000; //configure the output pins
    TRISAbits.TRISA4=1;    //configure the input pin
    CNEN1bits.CN0IE=1;    //enable change-on-notification CN0 interrupt
    CNPU1bits.CN0PUE=1;    //enables the weak pull-ups for CN0
    INTCON2 = 0x0001;    //on falling edge and set up INT0 pin to interrupt
                        //on rising edge
    IPC4bits.CNIP=2;    //configure the priority level for CN
    IFS1bits.CNIF = 0;    //Reset CN interrupt flag
    IEC1bits.CNIE = 1;    //Enable CN Interrupt Service Routine

    ConfigUART();
    ConfigADC();
    ConfigSPI();

}

unsigned int ReadADC12(unsigned char bufIndex)
{
    return(*(&ADC1BUF0+bufIndex)); //read ADC value
}

unsigned int processamento(unsigned int med, float G, long off)
{
}

unsigned int pO2;
long aux;

float xx;

// %O2=exp((med*10-off)/G)
aux=(long)((long)med*(long)10);
aux=aux-off;
xx=(float)((float)(aux)/G);
pO2=(exp(xx));
return pO2;
}

unsigned int AdcMean16()
{
    unsigned int media, i, ADC_VALUE;
    ConfigTMR(0x00000200); //configuration sampling frequency
    AD1CON1bits.ADON=1;    //activated ADC module
    media=0;
    for(i=1;i<17;i++)    //digital-filter with a mean of 16 ADC values
    {
        while(IFS0bits.AD1IF==0);
        IFS0bits.AD1IF = 0;
        ADC_VALUE=ReadADC12(0);
        media=ADC_VALUE+media;
    }
    media=media>>4;
    return media;
}

```


ANEXO C – Programa para armazenamento de dados na memória Flash do PIC em tempo de execução.

```
/******
* © 2005 Microchip Technology Inc.
*
* FileName:      rtsApi.s
* Dependencies:  Header (.h) files if applicable, see below
* Processor:     dsPIC33Fxxx/PIC24Hxxx
* Compiler:      MPLAB® C30 v3.00 or higher
*
*
* SOFTWARE LICENSE AGREEMENT:
* Microchip Technology Incorporated ("Microchip") retains all ownership and
* intellectual property rights in the code accompanying this message and in all
* derivatives hereto. You may use this code, and any derivatives created by
* any person or entity by or on your behalf, exclusively with Microchip's
* proprietary products. Your acceptance and/or use of this code constitutes
* agreement to the terms and conditions of this notice.
*
* CODE ACCOMPANYING THIS MESSAGE IS SUPPLIED BY MICROCHIP "AS IS". NO
* WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED
* TO, IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A
* PARTICULAR PURPOSE APPLY TO THIS CODE, ITS INTERACTION WITH MICROCHIP'S
* PRODUCTS, COMBINATION WITH ANY OTHER PRODUCTS, OR USE IN ANY APPLICATION.
*
* YOU ACKNOWLEDGE AND AGREE THAT, IN NO EVENT, SHALL MICROCHIP BE LIABLE, WHETHER
* IN CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE OR BREACH OF STATUTORY DUTY),
* STRICT LIABILITY, INDEMNITY, CONTRIBUTION, OR OTHERWISE, FOR ANY INDIRECT, SPECIAL,
* PUNITIVE, EXEMPLARY, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, FOR COST OR EXPENSE OF
* ANY KIND WHATSOEVER RELATED TO THE CODE, HOWSOEVER CAUSED, EVEN IF MICROCHIP HAS BEEN
* ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT
* ALLOWABLE BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO
* THIS CODE, SHALL NOT EXCEED THE PRICE YOU PAID DIRECTLY TO MICROCHIP SPECIFICALLY TO
* HAVE THIS CODE DEVELOPED.
*
* You agree that you are solely responsible for testing the code and
* determining its suitability. Microchip has no obligation to modify, test,
* certify, or support the code.
*
* REVISION HISTORY:
* ~~~~~
* Author      Date      Comments on this revision
* ~~~~~
* Settu D      03/15/06  First release of source file
* ~~~~~
*
* ADDITIONAL NOTES:
*
* Last modification by Almerindo Oliveira Paiva aop@ua.pt at 6/04/2009
*****/

#ifdef __dsPIC33F
#include "p33fxxxx.inc"
#endif

#ifdef __PIC24H
#include "p24hxxxx.inc"
#endif

.equ FLASH_PAGE_ERASE_CODE, 0x4042
.equ FLASH_ROW_PROG_CODE, 0x4001

.global _flashPageRead
.global _flashPageErase
.global _flashPageWrite
.section .text

/******
Flash Page Read
Read EIGHT rows (PAGE) of memory, upper PM byte is discarded
*****/
_flashPageRead:
    push    TBLPAG
    mov     w0, TBLPAG
```

```

        mov    #64, w3

readNext:
    tblrdl    [w1],[w2++]
    tblrdh    [w1++],w6           ; Discard PM upper byte
    dec      w3, w3
    bra      nz, readNext
    clr      w0
    pop      TBLPAG
    return

/*****
Flash Page Erase
Erase EIGHT rows (PAGE) of memory
*****/
_flashPageErase:
    push     TBLPAG
    mov      w0, TBLPAG           ; Init Pointer to row to be erased
    tblwtl   w1,[w1]             ; Dummy write to select the row
                                ; Setup NVCON for page erase

    mov      #FLASH_PAGE_ERASE_CODE, w7
    mov      w7, NVMCON
    disi     #5                  ; Block all interrupt with priority <7 for next 5
                                ; instructions

    mov      #0x55, W0
    mov      W0, NVMKEY
    mov      #0xAA, W0
    mov      W0, NVMKEY
                                ; Start Program Operation
    bset     NVMCON, #WR
    nop
    nop

erase_wait:
    btsc     NVMCON, #WR
    bra      erase_wait
    clr      w0
    pop      TBLPAG
    return

/*****
Flash Page Program
Program EIGHT rows (PAGE) of memory,
Each row contains 192 bytes of data (64 instructions, with upper PM byte == NOP)
*****/
_flashPageWrite:
    push     TBLPAG
    mov      w0, TBLPAG           ; Init Pointer to row to be programmed
    mov      #0, w6
    mov      #64, w3

pinst_loop:
    tblwtl   [W2++],[W1]
    tblwth   w6, [W1++]           ; load 0x00 into 3rd byte (will be decoded as
                                ; NOP should PC attempt to execute)

    dec      w3, w3
    bra      nz, pinst_loop
                                ; Setup NVCON for row program
    mov      #FLASH_ROW_PROG_CODE, w7
    mov      w7, NVMCON
    disi     #5                  ; Block all interrupt with priority <7 for next 5
                                ; instructions

    mov      #0x55, W0
    mov      W0, NVMKEY
    mov      #0xAA, W0
    mov      W0, NVMKEY
                                ; Start Program Operation
    bset     NVMCON, #WR
    nop
    nop

prog_wait:
    btsc     NVMCON, #WR
    bra      prog_wait
    clr      w0
    pop      TBLPAG
    return

.end

```

ANEXO D – Ficheiro cabeçalho de Device-drivers

```
/******
```

```
* FileName: device_driversPIC24Hv5.h
```

```
* Author Almerindo Oliveira Paiva    Date 20/02/09
```

```
*****
```

```
*Last modification at 11/06/2009
```

```
*****/
```

```
#include <math.h>
```

```
#include "spi.h"
```

```
#include <math.h>
```

```
#include "adc.h"
```

```
#include "timer.h"
```

```
#include "uart.h"
```

```
#include <stdio.h>
```

```
void StartPIC();
```

```
void ConfigUART();
```

```
void ConfigTMR(unsigned long period);
```

```
void ConfigADC();
```

```
void ConfigSPI();
```

```
unsigned int processamento(unsigned int med,float m, long b);
```

```
unsigned int ReadADC12(unsigned char bufIndex);
```

```
unsigned int AdcMean16();
```

```
unsigned int ReadUartInt();
```

```
void Print1(unsigned int aux);
```

```
unsigned int Print2(unsigned int aux);
```

```
void PutChar(char Char);
```

```
void PutStr(char * st);
```

```
void PutInteger (unsigned int valInt);
```

ANEXO E – Ficheiro Cabeçalho do programa para armazenamento de dados na memória Flash

```
/******
* © 2005 Microchip Technology Inc.
*
* FileName:      rtsApi.h
* Dependencies:  Header (.h) files if applicable, see below
* Processor:     dsPIC33Fxxx/PIC24Hxxx
* Compiler:      MPLAB® C30 v3.00 or higher
*
* SOFTWARE LICENSE AGREEMENT:
* Microchip Technology Incorporated ("Microchip") retains all ownership and
* intellectual property rights in the code accompanying this message and in all
* derivatives hereto. You may use this code, and any derivatives created by
* any person or entity by or on your behalf, exclusively with Microchip's
* proprietary products. Your acceptance and/or use of this code constitutes
* agreement to the terms and conditions of this notice.
*
* CODE ACCOMPANYING THIS MESSAGE IS SUPPLIED BY MICROCHIP "AS IS". NO
* WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED
* TO, IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A
* PARTICULAR PURPOSE APPLY TO THIS CODE, ITS INTERACTION WITH MICROCHIP'S
* PRODUCTS, COMBINATION WITH ANY OTHER PRODUCTS, OR USE IN ANY APPLICATION.
*
* YOU ACKNOWLEDGE AND AGREE THAT, IN NO EVENT, SHALL MICROCHIP BE LIABLE, WHETHER
* IN CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE OR BREACH OF STATUTORY DUTY),
* STRICT LIABILITY, INDEMNITY, CONTRIBUTION, OR OTHERWISE, FOR ANY INDIRECT, SPECIAL,
* PUNITIVE, EXEMPLARY, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, FOR COST OR EXPENSE OF
* ANY KIND WHATSOEVER RELATED TO THE CODE, HOWSOEVER CAUSED, EVEN IF MICROCHIP HAS BEEN
* ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT
* ALLOWABLE BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO
* THIS CODE, SHALL NOT EXCEED THE PRICE YOU PAID DIRECTLY TO MICROCHIP SPECIFICALLY TO
* HAVE THIS CODE DEVELOPED.
*
* You agree that you are solely responsible for testing the code and
* determining its suitability. Microchip has no obligation to modify, test,
* certify, or support the code.
*
* REVISION HISTORY:
* ~~~~~~
* Author      Date      Comments on this revision
* ~~~~~~
* Settu D      03/15/06  First release of source file
* ~~~~~~
*
* ADDITIONAL NOTES:
*
* Last modification by Almerindo Oliveira Paiva aop@ua.pt at 6/04/2009
*****/
#include "typeDef.h"

#ifndef __RTSPAPI_H__
#define __RTSPAPI_H__

// Flash Memory is organised into ROWs of 64 instructions or 192 bytes
// RTSP allows the user to erase a PAGE of memory which consists of EIGHT ROWs (512 instructions or 1536bytes) at a time.
// RTSP allows the user to program a ROW (64 instructions or 192 bytes) at a time

/*
* FLASH PAGE ERASE
*
* Parameters Definition:
* nvmAdu:      Selects the upper 8bits of the location to program or erase in program flash memory
* nvmAdr:      Selects the location to program or erase in program flash memory
* It must be aligned to 512 instruction boundary, LSB 10bits of address must be zero
*
* Return Value:
* Function returns ERROREE (or -1), if it is not successful
* Function return ZERO, if successful
*/
extern int flashPageErase(u16 nvmAdu, u16 nvmAdr);
```

```

/*
 * FLASH PAGE READ
 *
 * Parameters Definition:
 * nvmAdru:      Selects the upper 8bits of the location to read in program flash memory
 * nvmAdr:  Selects the location to read in program flash memory
 *           It must be aligned to 512 instruction boundary, LSB 10bits of address must be zero
 * pageBufPtr: Pointer to the data array in which read data will be stored

 * Return Value:
 * Function returns ERROREE (or -1), if it is not successful
 * Function return ZERO, if successful
 */

```

```

extern int flashPageRead(u16 nvmAdru, u16 nvmAdr, i16 *pageBufPtr);

```

```

/*
 * FLASH PAGE MODIFY
 *
 * Parameters Definition:
 * row:      Selects the row in the the Flash page that will be modified
 * rowBuf:  Selects the location to read in program flash memory
 * pageBufPtr: Pointer to the page data array, that will be modified

 * Return Value:
 * Function returns ERROREE (or -1), if it is not successful
 * Function return ZERO, if successful
 */

```

```

extern int flashPageModify(u16 row, u16 size, ConstInMem *rowBuf, i16 *pageBufPtr);

```

```

/*
 * FLASH PAGE WRITE
 *
 * Parameters Definition:
 * nvmAdru:      Selects the upper 8bits of the location to program or erase in
 *               program flash memory
 * nvmAdr:  Selects the location to program or erase in program flash memory
 *           It must be aligned to 512 instruction boundary, LSB 10bits of
 *           address must be zero
 * pageBufPtr: Pointer to the data array that needs to be programmed

 * Return Value:
 * Function returns ERROREE (or -1), if it is not successful
 * Function return ZERO, if successful
 */

```

```

extern int flashPageModify(u16 row, u16 size, ConstInMem *rowBuf, i16 *pageBufPtr);

```

```

#endif

```

ANEXO F – Reserva de Memória Flash para os parâmetros da Sonda

```
/******
* © 2005 Microchip Technology Inc.
*
* FileName:        testData.h
* Dependencies:    Header (.h) files if applicable, see below
* Processor:       dsPIC33Fxxx/PIC24Hxxx
* Compiler:        MPLAB® C30 v3.00 or higher
*
* SOFTWARE LICENSE AGREEMENT:
* Microchip Technology Incorporated ("Microchip") retains all ownership and
* intellectual property rights in the code accompanying this message and in all
* derivatives hereto. You may use this code, and any derivatives created by
* any person or entity by or on your behalf, exclusively with Microchip's
* proprietary products. Your acceptance and/or use of this code constitutes
* agreement to the terms and conditions of this notice.
*
* CODE ACCOMPANYING THIS MESSAGE IS SUPPLIED BY MICROCHIP "AS IS". NO
* WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED
* TO, IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A
* PARTICULAR PURPOSE APPLY TO THIS CODE, ITS INTERACTION WITH MICROCHIP'S
* PRODUCTS, COMBINATION WITH ANY OTHER PRODUCTS, OR USE IN ANY APPLICATION.
*
* YOU ACKNOWLEDGE AND AGREE THAT, IN NO EVENT, SHALL MICROCHIP BE LIABLE, WHETHER
* IN CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE OR BREACH OF STATUTORY DUTY),
* STRICT LIABILITY, INDEMNITY, CONTRIBUTION, OR OTHERWISE, FOR ANY INDIRECT, SPECIAL,
* PUNITIVE, EXEMPLARY, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, FOR COST OR EXPENSE OF
* ANY KIND WHATSOEVER RELATED TO THE CODE, HOWSOEVER CAUSED, EVEN IF MICROCHIP HAS BEEN
* ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT
* ALLOWABLE BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO
* THIS CODE, SHALL NOT EXCEED THE PRICE YOU PAID DIRECTLY TO MICROCHIP SPECIFICALLY TO
* HAVE THIS CODE DEVELOPED.
*
* You agree that you are solely responsible for testing the code and
* determining its suitability. Microchip has no obligation to modify, test,
* certify, or support the code.
*
* REVISION HISTORY:
* ~~~~~~
* Author      Date      Comments on this revision
* ~~~~~~
* Settu D      03/15/06  First release of source file
* ~~~~~~
*
* ADDITIONAL NOTES:
*
* Last modification by Almerindo Oliveira Paiva aop@ua.pt at 6/04/2009
*****/

#ifndef __TESTDATA_H__
#define __TESTDATA_H__
#include "typeDef.h"

#define PM_ROW __attribute__((space(prog), address(0x1000), aligned(128)))
const PM_ROW MyRowDataInFlash[] =
{
    0x9d19,0xc61e,0xd566,0x0000,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3353,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,
    0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333,0x3333
};

#endif
```

ANEXO G – Criação de novos tipos de variáveis utilizados pelos ficheiros dos Anexos A e B

/******

* © 2005 Microchip Technology Inc.

*

* FileName: typeDef.h

* Dependencies: Header (.h) files if applicable, see below

* Processor: dsPIC33Fxxx/PIC24Hxxx

* Compiler: MPLAB® C30 v3.00 or higher

*

* SOFTWARE LICENSE AGREEMENT:

* Microchip Technology Incorporated ("Microchip") retains all ownership and

* intellectual property rights in the code accompanying this message and in all

* derivatives hereto. You may use this code, and any derivatives created by

* any person or entity by or on your behalf, exclusively with Microchip's

* proprietary products. Your acceptance and/or use of this code constitutes

* agreement to the terms and conditions of this notice.

*

* CODE ACCOMPANYING THIS MESSAGE IS SUPPLIED BY MICROCHIP "AS IS". NO

* WARRANTIES, WHETHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED

* TO, IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A

* PARTICULAR PURPOSE APPLY TO THIS CODE, ITS INTERACTION WITH MICROCHIP'S

* PRODUCTS, COMBINATION WITH ANY OTHER PRODUCTS, OR USE IN ANY APPLICATION.

*

* YOU ACKNOWLEDGE AND AGREE THAT, IN NO EVENT, SHALL MICROCHIP BE LIABLE, WHETHER

* IN CONTRACT, WARRANTY, TORT (INCLUDING NEGLIGENCE OR BREACH OF STATUTORY DUTY),

* STRICT LIABILITY, INDEMNITY, CONTRIBUTION, OR OTHERWISE, FOR ANY INDIRECT, SPECIAL,

* PUNITIVE, EXEMPLARY, INCIDENTAL OR CONSEQUENTIAL LOSS, DAMAGE, FOR COST OR EXPENSE OF

* ANY KIND WHATSOEVER RELATED TO THE CODE, HOWSOEVER CAUSED, EVEN IF MICROCHIP HAS BEEN

* ADVISED OF THE POSSIBILITY OR THE DAMAGES ARE FORESEEABLE. TO THE FULLEST EXTENT

* ALLOWABLE BY LAW, MICROCHIP'S TOTAL LIABILITY ON ALL CLAIMS IN ANY WAY RELATED TO

* THIS CODE, SHALL NOT EXCEED THE PRICE YOU PAID DIRECTLY TO MICROCHIP SPECIFICALLY TO

* HAVE THIS CODE DEVELOPED.

*

* You agree that you are solely responsible for testing the code and

* determining its suitability. Microchip has no obligation to modify, test,

* certify, or support the code.

*

* REVISION HISTORY:

~~~~~

\* Author        Date        Comments on this revision

~~~~~

* Settu D 03/15/06 First release of source file

~~~~~

\*

\* ADDITIONAL NOTES:

\*

\* Last modification by Almerindo Oliveira Paiva aop@ua.pt at 6/04/2009

\*\*\*\*\*/

#ifndef \_\_TYPEDEF\_H\_\_

#define \_\_TYPEDEF\_H\_\_

typedef unsigned int u16;

typedef signed int i16;

typedef unsigned long u32;

typedef signed long i32;

typedef struct {

float m;

long b;

char lixo[64-(sizeof(float))-sizeof(long)];

} ConstInMem;

#endif

**ANEXO H – Manual de utilizador**

# **Medidor da Percentagem de Oxigénio em Gases**

## **Manual de Utilizador**



## Índice

|                                                                                  |    |
|----------------------------------------------------------------------------------|----|
| 1. O que é o Medidor da Percentagem de Oxigénio em Gases? .....                  | 3  |
| 1.1. Conteúdos da embalagem: .....                                               | 3  |
| 1.2. Requisitos relativos ao PC e Sistema Operativo.....                         | 3  |
| 1.3. Especificações de funcionamento.....                                        | 3  |
| 2. Instalação .....                                                              | 4  |
| 2.1. Algumas considerações sobre o local de instalação da sonda.....             | 4  |
| 2.2. Ligação Sonda à placa PCB .....                                             | 4  |
| 2.3. Comunicação com o PC .....                                                  | 5  |
| 2.4. Transmissão dos valores de %O <sub>2</sub> pela saída analógica 4-20mA..... | 5  |
| 3. Operação .....                                                                | 6  |
| 3.1. Comunicação com o PC .....                                                  | 6  |
| 4. Calibração.....                                                               | 8  |
| 4.1. Esquema de Calibração .....                                                 | 8  |
| 4.2. Equipamento necessário:.....                                                | 9  |
| 4.3. Procedimento .....                                                          | 9  |
| Cuidados:.....                                                                   | 11 |

## 1. O que é o Medidor da Percentagem de Oxigénio em Gases?

O *Medidor da Percentagem de Oxigénio em Gases* é um instrumento que serve para medir directamente a percentagem de oxigénio presente num volume de gás. O seu funcionamento baseia-se na resposta da sonda lambda, desenvolvida inicialmente pelo fabricante de acessórios para a indústria automóvel *Bosch GmbH*, na primeira parte da década de 1970.

O *Medidor da Percentagem de Oxigénio em Gases* possui uma interface RS-232, para comunicar com um PC. Poderá usar um cabo conversor RS-232 – USB, no caso de o PC não possuir qualquer porta RS-232 disponível.

O *Medidor da Percentagem de Oxigénio em Gases* está preparado para ser integrado nos sistemas de controlo das caldeiras industriais e, como tal, possui uma saída analógica em corrente que varia linearmente, entre 4-20mA, com a percentagem de oxigénio, isto é, de acordo com a fórmula  $I = 4 + \frac{16}{25} * \%O_2$  (mA).

### 1.1. Conteúdos da embalagem:

- Uma sonda lambda LSM11 – tecnologia *Bosch*;
- Uma Placa PCB;
- Um Cabo RJ45 com ficha macho RJ45 e ficha fêmea DB9;
- Um Cabo conversor RS-232 – USB;
- CD-Drive para instalar os drivers necessários à utilização do cabo conversor RS-232 – USB.
- T Adaptador para Calibração.

### 1.2. Requisitos relativos ao PC e Sistema Operativo

- Um PC com o Sistema Operativo Windows XP possuindo o programa *HyperTerminal* (Iniciar->Acessórios->Comunicação->Hyperterminal). No caso de o PC possuir instalado o Sistema Operativo Vista, deverá instalar o programa *HyperTerminal Private Edition 7.0*, que pode ser obtido gratuitamente no endereço <http://www.brothersoft.com/hyperterminal-private-edition-download-71955.html>.

### 1.3. Especificações de funcionamento

- Temperatura máxima dos gases de combustão: 800°C
- Tensão de Alimentação: 220V 50Hz.
- Temperatura Ambiente de operação: 0-60°C

## 2. Instalação

Antes de começar a instalar a sonda, confirme se os conteúdos que vêm na embalagem estão de acordo com a lista acima indicada.

### 2.1. Algumas considerações sobre o local de instalação da sonda

Instale a sonda num ponto do circuito de gases de escape onde seja possível medir uma mistura representativa destes, e onde a temperatura da mistura seja tão quente quanto possível, mas que não exceda a temperatura máxima admissível.

Na medida do possível, instale a sonda na vertical (fios para cima).

Não instale a sonda próximo das saídas das chaminés, ou das entradas de ventilações de modo a evitar a contaminação pelo ar exterior (atmosférico).

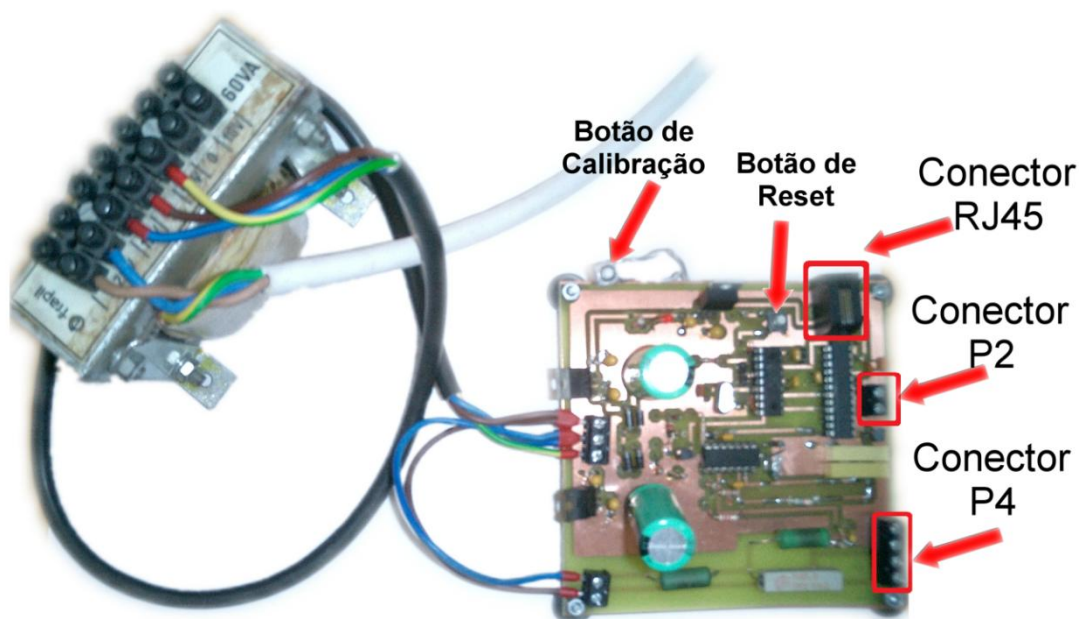


FIGURA 2-1 – MEDIDOR DA PERCENTAGEM DE OXIGÉNIO EM GASES

### 2.2. Ligação Sonda à placa PCB

Na placa existe uma ficha de ligação P4 para alimentar a sonda e para receber o sinal que esta fornece. Ligue, apertando os parafusos, os cabos da sonda ao conector P4 conforme ilustrado na figura 2.2. Tenha em atenção e respeite as cores dos fios.

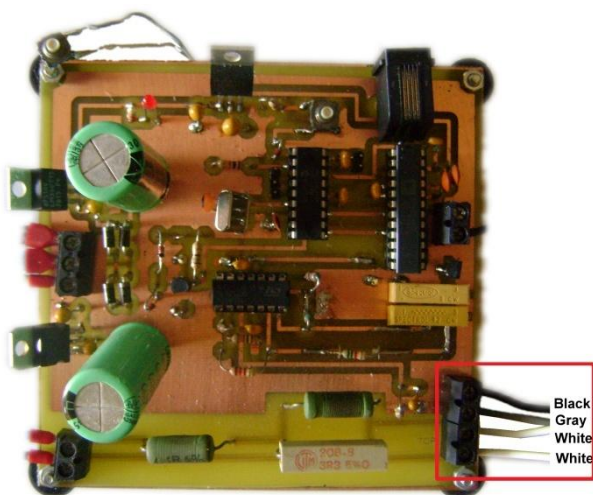


FIGURA 2-2 – LIGAÇÃO DOS CABOS DA Sonda À PLACA PCB

### 2.3. Comunicação com o PC

Na placa PCB existe um ligador RJ45 que permite a comunicação do Medidor da Percentagem de Oxigênio em Gases com um PC via RS-232. Para comunicar com o PC ligue o cabo RJ45 ao ligador RJ45 da placa PCB.

### 2.4. Transmissão dos valores de %O<sub>2</sub> pela saída analógica 4-20mA

Na placa PCB existe um ligador P2 que permite a comunicação dos valores de %O<sub>2</sub> através de um sinal de corrente analógico 4-20mA em modo *Active Loop*.

A figura 2.3 ilustra o modo como deverá utilizar este output na sua aplicação.

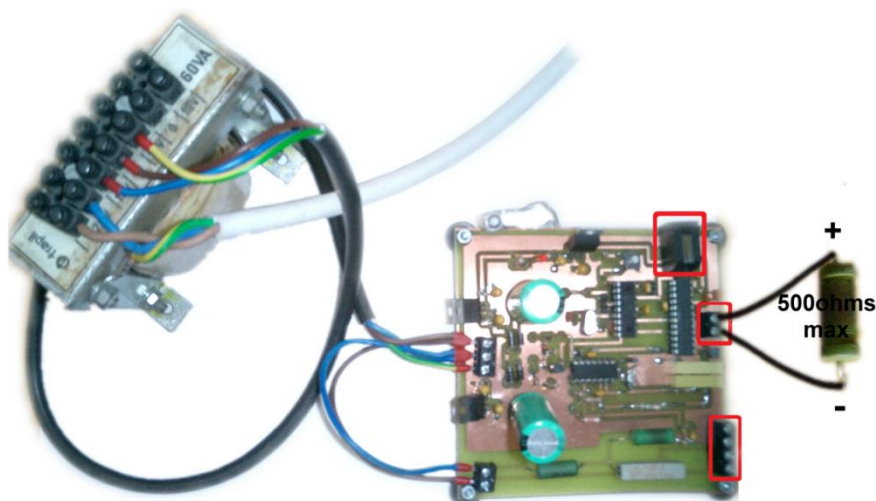


FIGURA 2-3– TRANSMISSÃO DE DADOS NO MODO *ACTIVE LOOP*

### 3. Operação

Após ter instalado a sonda num local apropriado, e depois de ter ligado a sonda à placa PCB e de ter estabelecido as ligações apropriadas com a saída analógica 4-20mA (caso esta saída seja utilizada na sua aplicação) poderá ligar a ficha do transformador à rede eléctrica. Verifique que entretanto o led vermelho acendeu.

Os valores transmitidos pelo *Medidor da Percentagem de Oxigénio em Gases*, tanto pela saída analógica como pela comunicação RS-232 só são validos 5 minutos depois de ligar o *Medidor da Percentagem de Oxigénio em Gases* à corrente eléctrica. Durante esse período a sonda lambda entra numa fase de aquecimento e os valores indicados por esta só serão válidos depois de o aquecimento estabilizar.

#### 3.1. Comunicação com o PC

O PC é utilizado tanto para monitorizar (ou ler) os valores de %O<sub>2</sub> de uma mistura gasosa como para calibrar a sonda.

##### 3.3.1. Monitorização dos valores de %O<sub>2</sub> em tempo-real no Hyperterminal ou programa equivalente

Ligue o cabo RJ45 ao conector RJ45 da placa PCB. Ligue a ficha DB9 a uma porta RS-232 ou utilize o cabo conversor RS-232 – USB e ligue este à ficha DB9 e à porta USB do PC.

Alimente a placa PCB. Entretanto abra o programa de computador *Hyperterminal* (*Windows XP*) que previamente instalou no PC no caso de utilizar o *Windows Vista*.

No campo **Name** introduza um nome para a comunicação com o *Medidor da Percentagem de Oxigénio em Gases* (p. e. Oxygen) e clique em **Ok**.

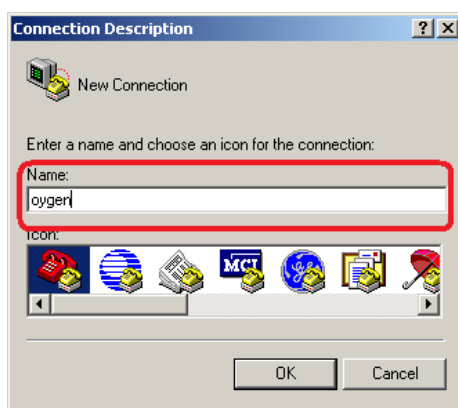


FIGURA 3-1- CRIAÇÃO DE UMA JANELA HIPERterminal

Selecione a porta COM que fará a comunicação com o *Medidor da Percentagem de Oxigénio em Gases* no campo **Connect using**.



FIGURA 3-2 – SELECÇÃO DA PORTA COM DO PC

Configure as propriedades da porta COM de acordo com os parâmetros indicados na figura seguinte.

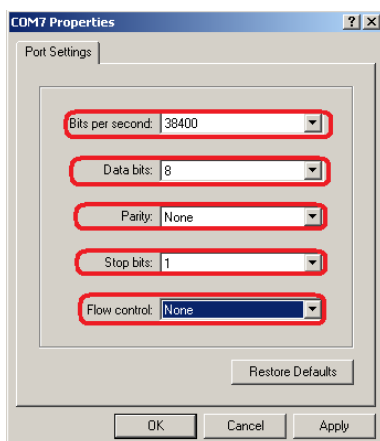


FIGURA 3-3 – CONFIGURAÇÃO DAS PROPRIEDADES DA PORTA COM

Clique em **Ok**.

Após estas configurações deverá surgir no ecrã do seu PC uma janela *Hyperterminal* a indicar os valores de percentagem de oxigénio dos gases que a sonda está a monitorizar.

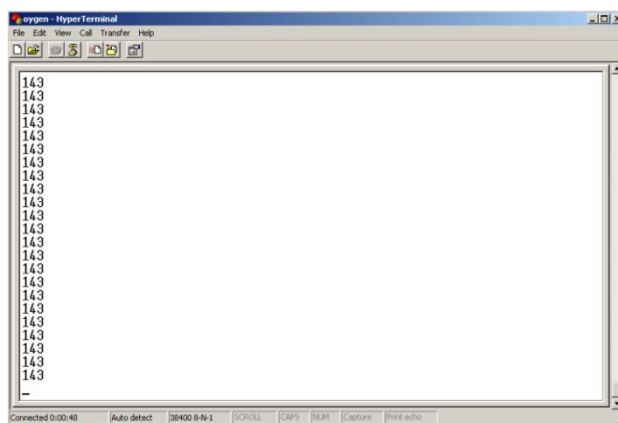


FIGURA 3-4 – INDICAÇÃO DOS VALORES DA PERCENTAGEM DE OXIGÊNIO NO *HYPERTERMINAL*

### 3.3.2. Calibração

O processo de calibração irá ser explicado na secção 4.

## 4. Calibração

O *Medidor da Percentagem de Oxigénio em Gases* dispõe de um método de calibração automático, simples e relativamente rápido.

### 4.1. Esquema de Calibração

A figura 4.1 ilustra o esquema de montagem mecânica a utilizar para calibrar a sonda.

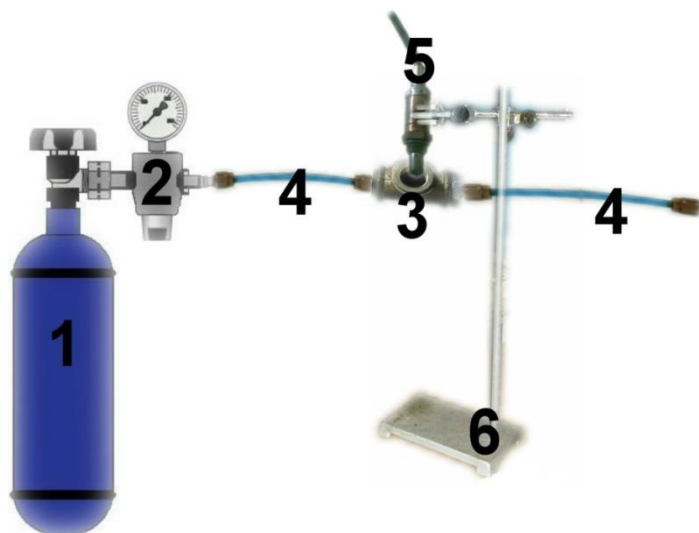


FIGURA 4-1 – ESQUEMA DE MONTAGEM DO PROCESSO DE CALIBRAÇÃO

#### 4.2. Equipamento necessário:

- 2 Botijas com os gases de calibração (ou apenas 1 botija de ar comprimido no caso de utilizar o ar atmosférico como segundo gás de referencia); (1)
- 1 Manómetro (Válvula redutora) para garantir que a pressão durante a calibração é inferior a 50mm de coluna de agua; (2)
- Adaptador de calibração; (3)
- Tubagem; (4)
- Sonda Lambda; (5)
- Suporte. (6)

#### 4.3. Procedimento

Para calibrar o *Medidor da Percentagem de Oxigénio em Gases*, proceda do seguinte modo:

1. Desligue a tomada do Medidor da Percentagem de Oxigénio em Gases.
2. Aguarde cerca de 10 minutos para deixar arrefecer a sonda lambda.
3. Retire a sonda lambda do local onde foi aplicada e insira esta no T adaptador de calibração.
4. Ao T adaptador de calibração ligue a tubagem a jusante e a montante da sonda lambda, como mostra a figura 4.1.
5. Coloque o adaptador de calibração num suporte de modo a que a cablagem eléctrica da sonda esteja voltada para cima.
6. Ligue a entrada da válvula redutora à garrafa de gás de calibração e a saída à entrada do calibrador.
7. Ligue o cabo RJ45 à ficha RJ45 da placa PCB. Ligue a ficha DB9 a uma porta RS-232 ou utilize o cabo conversor RS-232 – USB e ligue este à ficha DB9 e à porta USB do PC.
8. Alimente com tensão a placa PCB e aguarde cerca de 5 minutos até a sonda aquecer.
9. Abra a garrafa de gás e ajuste a válvula redutora de modo a que a pressão dentro das tubagens seja inferior a 5cm de coluna de água.
10. Abra o programa *Hyperterminal* conforme indicado na secção 3.1.1.
11. Clique no botão de Calibração.
12. No *Hyperterminal* aparecerá uma mensagem a pedir uma confirmação para prosseguir com a calibração.



FIGURA 4-2 – FASE 1 DE CALIBRAÇÃO



14. Prima 'y' ou 'Y' no caso de pretender calibrar a sonda ou então prima 'n' ou 'N' caso pretenda sair.

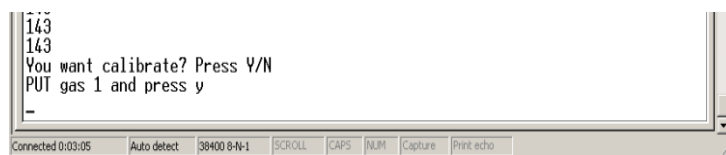


FIGURA 4-3 – FASE 2 DE CALIBRAÇÃO

15. Exponha a sonda a monitorizar o primeiro gás de calibração e prima 'y' ou 'Y'.

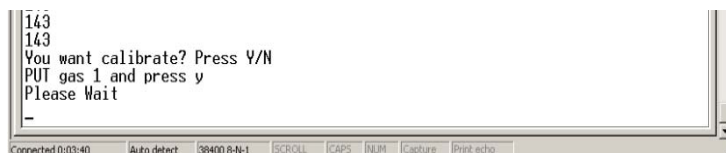


FIGURA 4-4 – FASE 3 DE CALIBRAÇÃO

16. Aguarde cerca de 2 minutos para garantir que a resposta da sonda seja estável.



FIGURA 4-5 – FASE 4 DE CALIBRAÇÃO

17. Introduza a percentagem de oxigénio do primeiro gás de calibração. **Note** que deve premir em primeiro o dígito da dezena, depois o dígito da unidade e por fim o dígito da parte decimal.



FIGURA 4-6 – FASE 5 DE CALIBRAÇÃO

18. Exponha a sonda a monitorizar o segundo gás de calibração e prima 'y' ou 'Y'.

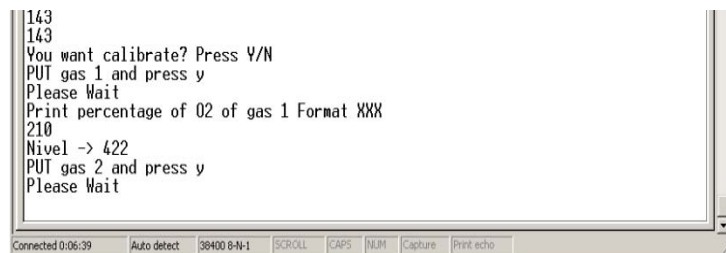


FIGURA 4-7 – FASE 6 DE CALIBRAÇÃO

19. Aguarde cerca de 2 minutos para garantir que a resposta da sonda seja estável.

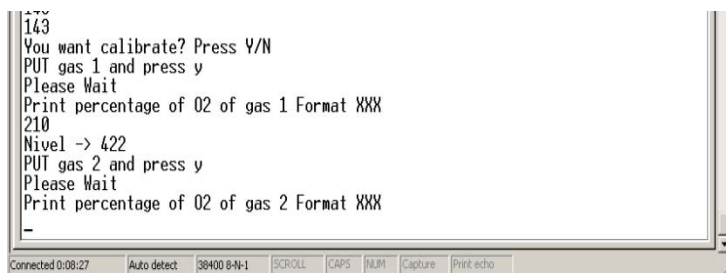


FIGURA 4-8 – FASE 7 DE CALIBRAÇÃO

20. Introduza a percentagem de oxigénio do segundo gás de calibração. **Note** que deve premir em primeiro o dígito da dezena, depois o dígito da unidade e por fim o dígito da parte decimal.

**Nota 1:** no caso de utilizar o ar atmosférico como gás de calibração, deve ligar a extremidade da tubagem a jusante da sonda a uma bomba aspirante de ar, e deixar a extremidade da tubagem a montante da sonda em aberto de modo a entrar ar atmosférico por esta.

**Nota 2:** Pode abortar a operação de calibração a qualquer instante pressionando o botão de *Reset*.

## Cuidados:

- Devido ao princípio de funcionamento da sonda, quando ligada está quente e pode causar queimaduras graves se não tomar as devidas precauções.
- A sonda recebe o gás de referência pelo mesmo orifício em que entra a cablagem, como tal, mantenha a cablagem sempre seca e limpa.
- Na limpeza da sonda, não utilize sprays de contacto nem agentes anti-corrosivos.
- A sonda deve ser ventilada de modo a evitar o sobreaquecimento.
- Proteja o *Medidor da Percentagem de Oxigénio em Gases* da humidade.